



## **neural networking**

- a new field of artificial intelligence

## **D.I.Y. toolkit**

serial mouse interrupts

## **new user guide**

concepts begin!

## **QL scene**

new London Quanta venue





# Fed up of DIGITAL PRECISION telling you how very good their software is?

✓ "As you might surmise by this time, I am impressed by QMATHS's abilities. Having noted that DIGITAL PRECISION's advertising tends to be loaded with superlatives (incredible, ultimate, superb come to mind), I had approached this evaluation with some scepticism. That scepticism has vanished." > INTERNATIONAL QL REPORT (IQLR, available from Miracle Systems) May/June 1993 issue, Official Review by M.Laverne commissioned by IQLR (who bought their review copy of the program).

✓ "PERFECTION is an exciting, full-flavoured, general purpose word processor of incredible capacity... PERFECTION has now been outshone by the recently released PERFECTION SPECIAL EDITION... The discoveries began to trip over themselves as PERFECTION SE responded to the keyboard with unexpected speed and intelligence... PERFECTION SE is blindingly fast at most things, and you are never left waiting for it. PERFECTION is everything that Quill never became: easy to use, very flexible, loaded with genuinely useful features, cleanly multi-tasking, capacious and incredibly fast. The SPECIAL EDITION offers 12 cylinder power and luxury to an already impressive package." > SINCLAIR QL WORLD magazine Official Review, April 1993 issue, by THE Mike Lloyd of Keyword Index / New QL User Guide fame.

✓ "I find PROFESSIONAL PUBLISHER an outstandingly good program that really does allow highly professional documents to be produced. For your interest I have included a few samples of work done for school using a combination of PERFECTION, PROFESSIONAL PUBLISHER, QUICKLASER and EYE-Q. You will be pleased to know that the quality has been rated so highly that people do not believe it can really have been done with just a QL... I must stress that I am already highly impressed with, and very satisfied by, the performance of PROFESSIONAL PUBLISHER and all the other DP programs that I use... I seem to learn something new that can be done almost each time I use the program. Very many thanks for helping to keep the QL ahead of the field." > Martin J Neave, Headteacher, Watton County Jnr School, Brandon Rd, Watton, Norfolk, IP25 6AL (unsolicited letter dated 18 May 1993 ordering more programs: Mr Neave had paid full price for everything).

✓ "LIGHTNING SPECIAL EDITION accelerates QL operation as nothing else does... more than 10x is achievable and 2x-4x is typical... I could not fault LIGHTNING SPECIAL EDITION on anything. It is a clear winner and a best buy at £49.95." > SINCLAIR QL WORLD magazine Official Review, April 1990 issue, by Ron Massey, who wrote EDITOR (bought full price) was "Superb" in an earlier review.

✓ "PERFECTION is well named" > R.H.Petford, Kingston Hill, Surrey, KT2 7LJ (unsolicited letter received May 25, 1993: another full price purchaser & upgrader).

✓ "When my ideal program finally arrived in the form of PROFESSIONAL PUBLISHER, it surpassed all my expectations... PROFESSIONAL PUBLISHER (is) in a class of its own, and makes it the only QL desktop publishing program for the very serious user... Until Digital Precision released PROFESSIONAL PUBLISHER, my opinion was that the use I could make of desktop publishing was mainly restricted to short documents... PROFESSIONAL PUBLISHER is a very versatile program... The illustrations for this series of articles have all been produced on PROFESSIONAL PUBLISHER... My printer is a BROTHER 9-pin dot matrix printer. It does illustrate the very high quality that can be obtained from PROFESSIONAL PUBLISHER even when using a simple printer." > SINCLAIR QL WORLD magazine Guide to desktop publishing ("A Question of Dots"), January 1992 to December 1992 issues: the reviewer had bought PROFESSIONAL PUBLISHER, PERFECTION SE, FONT ENLARGER, TOOLBOXES, QUICKLASER etc from Digital Precision all at full price.

✓ "I am aware that over the years Digital Precision has given considerable support to the QL scene but seldom, if ever, can there have been such estimable service as I recently encountered with PERFECTION PLUS." > The Hon. W.D.R. Spens, Bridgewater, Somerset, TA5 1HG, QUANTA magazine, March 1992 issue. Mr Spens has bought a lot of his software from Digital Precision, all at full price of course.

✓ "The Digital Precision Desktop Publisher was rightly hailed as an extraordinary programming achievement when it was released two years ago. Mike Lloyd casts a professional eye over Digital Precision's latest page-making blockbuster (PROFESSIONAL PUBLISHER) and finds plenty to be pleased about... there is unlikely to be a single program of such magnitude and quality (as PROFESSIONAL PUBLISHER) written for the Sinclair QL." > SINCLAIR QL WORLD Official Review, August 1989 issue, by M.Lloyd, who personally bought all this at full price.

✓ "EDITOR is a liberation. After Quill, it was like jumping from an aquarium into the sea. It has become part of my professional life... Everyone is now writing about the excellence of PERFECTION. I have not tried it, not having any perceived need for it (having EDITOR)" > Suzanne Cronje, QUANTA magazine, May 1992 issue, page 2. Ms Cronje naturally had paid the full price for her copy of EDITOR SE etc.

✓ "I have found (PERFECTION) to be simply excellent, fast, packed with features and very well thought out. I can find little to say that will convey just how good this program is, except to quote Digital Precision's own advertising: PERFECTION will blow your socks off. PERFECTION is the program that Quill users have been waiting for." > SINCLAIR QL WORLD's first Official Review, May 1991 issue.

✓ "Digital Precision (DP) decided to begin work on a replacement for Quill which would be very quick, simple to use and contain lots of excellent features - something upon which DP have built a very strong reputation in the QL market... Overall, the speed-up (of just the first release of PERFECTION - it is much faster now) on even a humble QL with Trump Card is amazing when compared with Quill (or any other word processor). On top of this, the program provides many excellent and well thought out features, each of which is easy to use... (it) is certainly years ahead of the competition on the QL (and even on many PCs)." > R.Mellor, c/o CGH Services, Cwm Gwen Hall, Pencader, Dyfed, SA39 9HA; Official Review of the very first version of PERFECTION in QL TECHNICAL REVIEW issue 7: and the reviewer personally bought his own copy of this program, and many others at full price, from Digital Precision. Earlier QLTR reviews pronounced LIGHTNING (just the standard version) superior to the competition and ADVENTURE CREATION TOOL excellent.

✓ "PERFECTION SE is superb!! With Gold Card, it puts life in the fast lane. Thanks." > Leonard Singleton, Bletchley, MK3 6BP, June 1993, a full price purchaser (=fpp).

✓ "As a recent user of PERFECTION PLUS SE, may I add my thanks and praises to the ones I am sure you have already received... keep up the excellent work." > R Slawson, East Molesey, Surrey, KT8 0BP (unsolicited letter from full price purchaser).

✓ "At about 360,000 words, the Mega SPELLCHECKER dictionary does not have much competition, on any computer! (Spellchecking) is about four times as fast as the best figures I have seen with other checkers on QL and PC." > SINCLAIR QL WORLD magazine official review of PERFECTION spellchecker, September 1992 issue, by Bryan Davies of Troubleshooter repute (review copies of all the competing products supplied to SINCLAIR QL WORLD by their respective publishers).

✓ "In the past I have purchased a number of your programs and have never failed to be impressed by the quality of both product and documentation. (So) please send a list of your current products." > V.Negri, Hempton, Norfolk, NR21 7LF, June 1993, fpp.

✓ "This is my first letter with PERFECTION SE. I must say I'm impressed with it and it is certainly fast. Hooray, goodbye to Archive!" > P.H.Heilbron, Reigate, RH2 0DJ, a full price purchaser now using PERFECTION to replace not only Quill but Archive too.

✓ "I have been using PROFESSIONAL PUBLISHER for about eighteen months now... what you can do with it is colossal... I got Digital Precision's QUICKLASER. The results are as good as (Digital Precision) says in its advertisements..." > P.Hamill, Peterborough, Cambs, PE8 6RH, QUANTA magazine, Volume 9 issues 4/12. Mr Hamill (full price purchaser) then makes suggestions to users re optimal page sizes.

✓ "Once again I would like to say thank you for your help. I would like to tell the world what nice guys you are but unfortunately I have no contact with the outside world." > J.Bailey, Godshill, Ventnor, PO38 3JJ (full price purchaser, 24 May 1993).

✓ "PC CONQUEROR GOLD SPECIAL EDITION is an excellent product, accompanied, as so often with Digital Precision software, by a comprehensive and informative manual. The program does a difficult job, and does it well... Overall, this program is much faster, more compatible and capable..." > SINCLAIR QL WORLD Official Review, March 1993 issue, by M.Knight (bought many DP programs full price).

✓ "With printing of the quality that this page bears witness to, I am a very satisfied PERFECTION user. I hope that you continue to provide the software innovation and the accessible backup which is great. So, thank you very much again and may I wish you every good fortune." > P.Stewart, Temple, London, EC4Y 9BE, 10 May 1993, fpp.

✓ "Many thanks for the update of PERFECTION SPECIAL EDITION. I am suitably impressed. Congratulations on producing the only word processor that I know that offers the best of all worlds as far as formatting is concerned. After Quill, PERFECTION is like a breath of fresh air." > Geoff Wicks, 1097HL Amsterdam, Netherlands (unsolicited letter dated 13 June 1993: all software including LIGHTNING PERFECTION SE, PRO PUBLISHER, CONQUEROR SE etc. purchased at full price).

✓ "All I can say about QMATHS is: WOW!" > Robin Wyke-Holloway, Salisbury, SP5 4WG (unsolicited letter received April 1993: Mr Holloway is a full price purchaser).

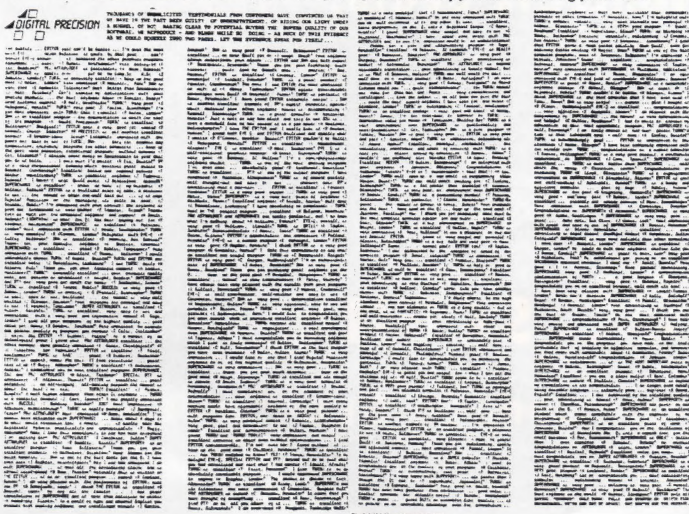
✓ "I have had PERFECTION from the early days and have had many hours pleasure finding out more and more of its brilliant features. May I offer my congratulations on such an easy to use program which does everything I want - and more besides." > F.Merrison, Pinner, HA5 5AZ, fpp, thanking us for fixing a printer problem he'd had.

✓ "Having used a range of desktop publishers on the Atari ST & Amiga, I admit I am very impressed with the superior performance of PROFESSIONAL PUBLISHER. It contains everything required" > SINCLAIR QL WORLD January 1989 issue, article entitled "Six of the Best" which also praised five other new Digital Precision programs.

✓ "I find it difficult to express my gratitude for the speed of your response, and for the opportunity to see inside two excellent programs which I have long enjoyed using." > Richard Walker, Enniskillen, BT74 7LG, full price purchaser and QL expert.

✓ "May I take this opportunity to say that I have, in the past, found the software you have supplied me with (LIGHTNING etc.) to be of extremely high standard, on a par with that found in industry-standard PC packages. Keep up the good work. Without your quality software, I would be forced to abandon the QL and go to a PC." > G. Reynolds, Crosby, Liverpool, L23 0SS (unsolicited letter dated April 2 1993, placing a further order for DP software: all programs old & new were purchased at full price).

✓ This is but a casual selection, drawing only on extracts from recent letters and reviews. We could locate >1000 complimentary communications but we'd prefer to spend our time producing new programs! We refer potential customers (existing users of DP products already know how good they are) to pages 18/19 of the September 1988 issue of Sinclair QL World, which contained well over three hundred other unsolicited quotations from happy Digital Precision customers. That collection covered only three programs (and that too only partly - we ran out of space) and predated the release of what many consider to be our best software (LIGHTNING SE, PERFECTION SE, PROFESSIONAL PUBLISHER, CONQUEROR SE etc). We reproduce those pages below, duly reduced to fit. If you want a readable copy, consult the relevant back issue or send us an SAE or ask for a full-sized copy while ordering from us...



## So hear it from their customers!



## SPECIAL DEALS

5% off total if you buy 2 programs/upgrades;  
10% off 3; 15% off 4; 20% off 5; 25% off 6+  
Upgrades cost difference in price + £10  
Non-UK Europe add 5%, rest of world 10%

For full terms and conditions, please refer to any of our QL World ads from Jan-Nov 1990, or write in including a SAE

## CPORT IMPROVED VERSION

A brand new CPORT system, enabling you to rapidly convert your SuperBASIC programs into C (ANSI or Lattice). The new (October 1992) version is now as close to being fully automatic as makes no difference – you must get it!

Owners of our earlier CPORT versions should return disk + SAE for a free upgrade.

## SUPERFORTH COMPILER WITH REVERSI

Forth is the most logical computer language. This compiler produces multitasking code. The manual teaches you Forth-83 from scratch.

## IDIS SPECIAL EDITION IDIS

These intelligent disassemblers make the otherwise terrifyingly complex task of understanding other people's machine code programs absurdly easy. The SE version, which has a higher hardware requirement, sorts out some routines, replaces addresses with names, untangles data from code and much more.

## QKICK FRONT END SYSTEM

This is a simple, easy-to-master, pull-down menu controlled multitasking front end. QKICK runs in the background and can be called up at any time. It provides you with notepads, sophisticated file/sector/RAM handling, backing up facilities, a clock, diary, calculator, mini-database and so on.

## ADVENTURE CREATION TOOL SPECIAL EDITION

ACT is a must for every programmer. The name of the program is misleading, insofar as it has capabilities far beyond the 'mere' creation of adventures. ACT has utilities providing animated graphics, data compression, language design, parsing, maps, object-oriented control etc. If all you want to do is generate adventures, though, you do not need to be a programmer to use it. This is a purchase you will never regret.

## PEDIT

A fast, modern and capable printer driver for the programs bundled with the QL.

## MICROBRIDGE

Superb contract bridge bidder (ACOL etc) and player, using millions of random but reconstructable hands. Microbridge also includes a state of the art interactive bidding tutor and a clear instruction manual. There is nothing like this anywhere else!

## SUPER ASTROLOGER

A very cut-down version of Professional Astrologer – still great fun, though!

## SUCCESS CP/M EMULATOR

Allows your QL to run CP/M programs at great speed.

## 3-D PRECISION CAD SYSTEM

ADVENTURE CREATION TOOL SPECIAL EDITION

ARCADIA GAME

ARCHIVE DEVELOPMENT SYS + RUN-TIME MODULE

ARCHIVE TUTORIAL

BETTER BASIC EXPERT SYSTEM

BLOCKLANDS GAME

CASH TRADER v3.3 + ANALYSER

COMPARE FILE UTILITY

COPY UTILITY FOR BACKING UP

CPORT BASIC TO C TRANSLATOR WITH C68K COMPILER

CPORT BASIC TO C TRANSLATOR

DATABASE ANALYSER FOR ARCHIVE

DAT-APPOINT APPOINTMENT DATABASE SYSTEM

DESKJET/LASERJET DRIVER FOR PRO PUBLISHER

DESKTOP PUBLISHER SPECIAL EDITION

DESKTOP PUBLISHER

DIGITAL C COMPILER

DIGITAL C SPECIAL EDITION

DISKTOOL WITH QUICKDISK

DROIDZONE GAME

DR-DOS v6.0 STATE OF THE ART PC OPERATING SYSTEM

EDITOR SPECIAL EDITION

EDITOR

EYE-Q GRAPHICS SYSTEM

FONT ENLARGER

GAMES COMPENDIUM (ALL FIVE GAMES)

GRAFIX 24-PIN UNIVERSAL PRINTER DRIVER

HARDBACK + FINDER HARD DISK MANAGER

IDIS INTELLIGENT DISASSEMBLER

IDIS SPECIAL EDITION

LIGHTNING ACCELERATOR

LIGHTNING SPECIAL EDITION ACCELERATOR

LIGHTNING SPECIAL EDITION FOR GOLD CARD

MAILMERGE DATABASE SYSTEM

MEDIA MANAGER SPECIAL EDITION

MEDIA MANAGER

MEGA DICTIONARY

MICROBRIDGE

NAMES AND ADDRESSES DATABASE SYSTEM

PAYROLL SYSTEM

PC CONQUEROR GOLD SPECIAL EDITION WITH DR-DOS v6.0

PC CONQUEROR GOLD SPECIAL EDITION

PC CONQUEROR WITH DR-DOS v6.0

PC CONQUEROR

PEDIT PRINTER DRIVER FOR XCHANGE

PERFECTION PLUS SPECIAL EDITION WITH SPELLCHECKER

PERFECTION PLUS WITH SPELLCHECKER

PERFECTION SPECIAL EDITION WORD PROCESSOR

PERFECTION WORD PROCESSOR

PERFECT POINTER TOOLS

PROFESSIONAL ASTROLOGER WITH ASTRONOMER

PROFESSIONAL ASTROLOGER

PROFESSIONAL ASTRONOMER

PROFESSIONAL PUBLISHER TOOLBOXES (PARTS ONE & TWO)

PROFESSIONAL PUBLISHER TOOLBOX PART ONE

PROFESSIONAL PUBLISHER TOOLBOX PART TWO

PROFESSIONAL PUBLISHER

QFLICK CARD INDEX SYSTEM

QKICK FRONT END SYSTEM

QMATHS MATHEMATICAL SYSTEM PART ONE

QMATHS MATHEMATICAL SYSTEM PART TWO

QMATHS MATHEMATICAL SYSTEM (PARTS ONE & TWO)

QMON MACHINE CODE MONITOR v2.05

RECOVER ARCHIVE UTILITY

REVERSI GAME

REDIT + SCREENPRINT ARCHIVE UTILITIES

SOLUTION PC EMULATOR WITH DR-DOS v6.0

SOLUTION PC EMULATOR

SPELLCHECKER WITH DICTIONARIES

SUCCESS CP/M EMULATOR

SUPERBASIC MONITOR

SUPERFORTH COMPILER WITH REVERSI

SUPER ASTROLOGER

SUPER BACKGAMMON GAME

SUPER SPRITE GENERATOR

TOOLKIT III WITH ROM

TOOLKIT III

TRANSFER UTILITY SPECIAL EDITION

TRANSFER UTILITY

TURBO BASIC COMPILER + TOOLKIT

ULTRAPRINT SCREEN DUMP UTILITY

XREF SUPERBASIC PROGRAM ANALYSER

KEY>>>>>> Available either on cartridge or disk

Available only on disk

Minimum 512K exp:only available on disk

Minimum 256K exp:either cartridge or disk

Minimum 256K exp:only available on disk

Available only on cartridge

Minimum 1.5Mb RAM:only available on disk

As well as cartridge or disk, you get a ROM



Send to: DIGITAL PRECISION LTD, 222 THE AVENUE, CHINGFORD, LONDON E4 9SE or phone 081-527 5493

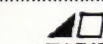
Please rush me: .....

Name: ..... Address: ..... Postcode: .....

Encl: CHEQUE/VISA/ACCESS/MASTERCARD/PO/MO/CASH for £ .....

Card No: ..... Expiry: ..... Signed: .....

Delete as appropriate: (QL/THOR/STQL)(MDV/3.5"/5.25")(Disk=360/720/1440/2880/6400 sectors; nb 1 sector=0.5K)(If 720 sectors: SSDD/DSSD)

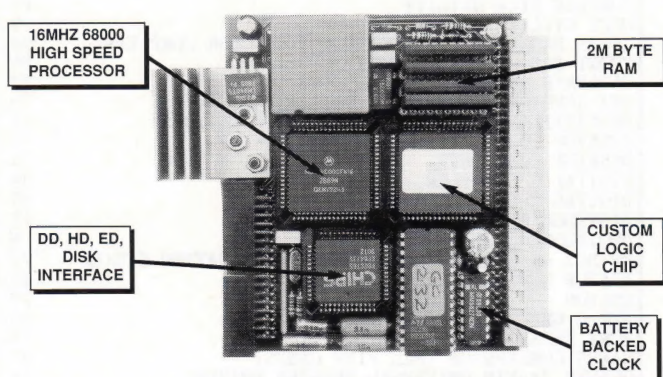


DIGITAL PRECISION LTD

DIGITAL PRECISION LTD



# MIRACLE SYSTEMS



## QL GOLD CARD

**£225 inc. (£200 outside EC)**

This is the expansion that has been revolutionising the QL. It is very easy to fit - it simply plugs into the expansion port at the left hand of the QL - and once fitted it will instantly increase the execution speed of the QL by about 4 times due to the presence of a 16MHz 68000 on board. There is 2M of fast 16 bit RAM of which QDOS sees a contiguous 1920K. The remainder is used for shadowing the QL's ROM and display memory and for the GOLD CARD's own code.

There is a disk interface which can access 3 mechanisms (4 with the DISK ADAPTER) of 3 different densities, DD (double density, 720K, HD (high density, 1.44M) and ED (extra high density, 3.2M) in any mix. The disk interface connector is the same type that was fitted to the TRUMP CARD so most QL compatible disk drives can be used. Please note that DD drives still give a capacity of 720K per diskette. Our DUAL ED DISK DRIVE allows the GOLD CARD to access DD, HD and ED diskettes.

Another feature is the battery backed clock. When the QL is switched on the contents of the clock are copied into the QL's clock so that the time and date are correct. The firmware in the ROM gives the GOLD CARD all the functionality of the TRUMP CARD like TOOLKIT II and there is a sub-directory system for floppy and RAM disks.

Physically the GOLD CARD is about half the size of the TRUMP CARD and so fits almost all within the QL. Its current consumption is well under the allowable maximum so no special power supply is required. The GOLD CARD comes with a 14 day money back guarantee and a 2 year warranty.

**See us here:**

**Sunday 23rd January 1993**

Georgian House Hotel  
Blackrod  
nr. Manchester

**Saturday 12th February 1993**

Scottish Workshop  
Edinburgh

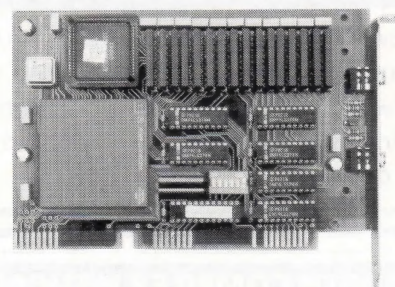
For venue details contact Alan Pemberton,  
68 Lingerwood Road, Newtongrange,  
Midlothian EH22 4QQ

**Tel: (0904) 423986**

**To place an order by phone please have your credit card ready. For overseas customers we charge the prices shown in brackets.**

**To order by post, please fill in the form opposite or write to us quoting your credit card number and expiry date, or enclosing a cheque payable to MIRACLE SYSTEMS Ltd.**

# THE QXL



The QXL turns the common PC into a QL compatible. The package comprises a half card that plugs into an 8 or 16 bit standard ISA slot and a diskette loaded with a QDOS compatible operating system and a Superbasic compatible interpreter. After installation simply type QXL and the PC will appear to be a QL allowing QL programs to be run from QL format diskettes.

The card itself has a 32 bit 68EC040 processor running at 20MHz which gives a good turn of speed. This processor has access to its own RAM and so performance is virtually independent of the host PC whether it has an 8088 or a Pentium. In fact the PC is used purely as an I/O system giving QL programs access to the PC's floppy disc, hard disc, keyboard, display, serial and parallel ports. The card itself has QL style network ports to allow connection to a QL network. The minimum PC specification required is an XT with EGA display and a spare standard slot.

Varying RAM sizes from 1M up to 8M can be supplied. The smaller capacities can be upgraded to the larger ones and the cost is simply the price difference. Not all the RAM is available to the user programs; the 1M equates roughly with a TRUMP CARD QL memory size and the 2M with a GOLD CARD QL.

During the lifetime of the QXL we intend to enhance the software to make use of the new hardware facilities of the PC such as SVGA graphics. As has been our policy with the TRUMP CARD and GOLD CARD we intend to provide software upgrades free of charge.

## QXL prices

<b>1M</b>	<b>£295</b>	<b>(£255)</b>
<b>2M</b>	<b>£325</b>	<b>(£280)</b>
<b>5M</b>	<b>£410</b>	<b>(£355)</b>
<b>8M</b>	<b>£495</b>	<b>(£430)</b>

(prices in brackets for outside EC)

**INTERNATIONAL QL REPORT (IQLR)** is a regular magazine that all QL users should read. It has articles for the beginner, the advanced user and every one else in between. Also, the international flavour combined with low advertising rates makes it probably the best place to locate QL related items. IQLR is run by QL enthusiasts whose proud boast is that they have never been late with an issue. If you do not already get it then 'phone us now. One year's subscription for 6 issues to any European address is £22.00 and it's worth every penny. Subscribers elsewhere should contact SeaCoast Services, 15 Kilburn Court, Newport, RI 02840, U.S.A. direct.



**To: MIRACLE SYSTEMS Ltd, 25 Broughton Way,  
Osbaldwick, York, YO1 3BG, U.K. Tel: (0904) 423986**

Please send me \_\_\_\_\_

I enclose a cheque to the value of: £ \_\_\_\_\_

Or debit credit card

Expires   /  

Name \_\_\_\_\_ Signature \_\_\_\_\_

Address \_\_\_\_\_



# Contents

**Editor**

Helen Armstrong

**Publisher**

Mark Kasprovicz

**Advertising Manager**

Jim Peskett

**Creative Director**

John Stanley

**Graphic Artist**

Steve Billington

**Magazine Services**

Linda Miller,  
Frances Maxwell,  
Pauline Wakeling,  
Val Brandon

**Sinclair QL World,**  
**Published by Arcwind Ltd.**  
**The Blue Barn,**  
**Tew Lane, Wootton,**  
**Woodstock,**  
**Oxon. OX7 1HA**  
**Tel: 0993 811181**  
**Fax: 0993 811481**  
**ISSN 026806X**

If you have any comments or difficulties please write to the editor and we will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies.

Back issues are available from the publisher price £2.50 UK, £2.99 Europe. Overseas rates on request.

Subscriptions from: Arcwind  
The Blue Barn, Tew Lane,  
Wootton, Woodstock, Oxon.  
OX7 1HA  
UK: £23.40  
Europe: £32.90  
Rest of World: £40.90

Reprographic Services:  
Eclipse, Brook Street,  
Watlington, Oxon. OX9 5JH.  
Distributed by: Seymour Press  
Ltd., Windsor House, 1270  
London Road, Norbury,  
London, SW16 4DH

© 1993 ARCWIND LTD.  
Sinclair QL World is published 12  
times a year. All rights reserved.  
Reproduction in whole or in part  
without written permission is  
strictly prohibited. We welcome  
contributions. All material must be  
supplied with a SAE.

While all reasonable care is taken  
in compiling QL World, the  
publisher and its agents assume  
no responsibility in affects arising  
therefrom. Opinions expressed  
are those of the authors.

## 6 TROUBLESHOOTER

Bryan Davies looks at the LineDesign updates, and the Psion 3.

## 8 OPEN CHANNEL

Other languages ... community help ... stuck with cursor colour ... naval lead ... blind  
need computer help ... family tree search ... dates again!

## 10 QL SCENE

London Quanta new venue ... New front end from Hessler ... Miracle latest ... QL Prolog  
... Ergon simplifies pricing! ... French AGM.

## 11 SUPERBASIC IN ACTION

Simon Goodwin code-caches with the new Motorola chips

## 15 THE NEW USER GUIDE - part 29 and 30

The Keyword Index from WHEN to WINDOWS ... and the introduction to CONCEPTS.

## 19 DIY TOOLKIT

Simon Goodwin serially interrupts mice with assembler.

## 23 NEURAL NETWORKING

Dr. Steven Wilcox brings the latest in artificial intelligence to the QL

## 28 REVIEW: PROPUB TOOLBOXES

Bryan Davies looks at DP's ProPub add-on software.

## 31 MICRO ADS

## Coming Soon

A map of the WORLD ... QL CALENDAR ... the Fifth Italian SHOW.



**Bryan  
Davies looks  
at some  
updates, the  
Psion 3, and  
an unwell  
Thor.**

# **S H TROUBLE O T E R**

In issue 11.9, Dilwyn Jones gave some advice on the use of the SERMouse serial mouse kit - that you need to be careful not to hold down a mouse button when switching on the QL. This applies to 3-button mice which do not have a mouse-mode switch on them. There may be other variations on this theme, though. When used on a PC, my own Genius GM6 mouse defaults to Mouse Systems mode (3-button) and will only operate as a Microsoft Mouse (2-button) if (any) one of the three buttons is held down during the boot process. This caused sufficient annoyance to push me into buying another mouse, which has the Microsoft/Mouse Systems switch on it.

The same issue has a letter from Ron Stewart, suggesting a fix for a problem which can occur with the Abacus Amend command. This fix was demonstrated at our last local Quanta group meeting, and it certainly seemed to work. There is no guarantee that making the patch he gives will not cause some other trouble, but it is definitely worth trying.

## **File Finder**

In my review of File Finder ("FiFi")

some months ago, I commented that a text string could not be located in a file created in Professional Publisher. The program writer replied that the reason for this is that the text was not in the file as text, but as bit-mapped images. This brings up a point that could trouble average users - when is text not text? My own understanding of how text is contained in DTP (or

high-end WP) program files is conditioned by experience with other programs, in which text does remain as "itself".

Programmers obviously see files differently from the way users see them; one looks at the file as a structure, the other sees it as an end-product something that is readable as normal language on a piece of paper. In writing reviews, I look at programs as a user. What goes on internally, within a file, does not normally interest me. It is only when something goes wrong with a file that its internals need to be investigated.

## **LineDesign**

Very soon after the initial release, Belgian software developers Progs have started supplying beta-test copies of version 2.00 of LineDesign for comment. As they promised earlier, they are working hard to improve areas of the program which fell short of the general program standard. They have not wasted any time bringing out important changes.

The open way in which Progs speak of the difficulties of getting certain program functions to work acceptably will appeal to many users. Making the problems clear, and assuring users (and potential buyers) that fixes are in-hand, should reassure people for whom the program is a substantial investment. We have all suffered from programs which have not been sufficiently de-bugged before release, but we have rarely been given any information on what the programmers are doing to make their masterpieces usable.

In this case, we even have printed instructions "especially for the beta-testers"! The first thing that is apparent

when 2.00 is fired up is that it is appreciably faster than 1.08 as reviewed in 11.9. All aspects seem to have benefitted. Loading documents for the review, another improvement became obvious - if a font specified in the file being loaded is not currently available, you can select another from a drop-down list. Zooming into the displayed page is a simple process and quick with an average type of page. The maximum size to which the selected portion of the page can be blown up should satisfy most users. What will please anyone who has used bit-map editor programs is the unchanging nature of the images as they are magnified; lines are not all absolutely straight, of course, but the image looks much the same whatever size it is displayed at.

Conventional scroll bars have replaced the toolbar arrows. A feature which has not yet been implemented is Insert Bitmap. This will allow QL 4- and 8-colour screen dumps to be inserted in current LineDesign pages. It is intended that other (non-QL?) formats will be supported by this command in the future. The revised program deserves more detailed comment, but that will have to wait until some hours have been spent using the new and revised commands.

## **Text87 Too**

As would be expected, changes have been made to Text87 to reflect the upgrading of LineDesign. There have also been a number of changes not related specifically to that program. The version numbers have been changing frequently in recent months, and may have gone beyond the 4.4 that appears to be current as of mid-November.

## **Readers' Letters**

Peter Hamill wrote an enthusiastic letter about LineDesign, which he is using to good effect on a newsletter for his local sailing club. The newsletter is a combination print, part done with Publishers' Pack and part with Text87 alone. As with several "classic" DTP programs, you need to make the best use of two or three programs to get the results people expect these



days - text from a WP program, perhaps graphics from a pukka drawing program, and the combination of text and graphics, partly from LineDesign and partly from multiple passes through the printer. This procedure may sound complex, but it need not be that bad. LineDesign obviously is a drawing program, and the reason for suggesting the use of another graphics program in addition is that there are some good effects and "potted" images available from non-QL programs, and it is tempting to make use of that which has already been created. LineDesign has the ability to import certain other types of file, such as those from Adobe Illustrator. It is highly desirable to have plenty of "spare" time, though, but that is what microcomputing is all about, surely?

One tip Peter passes on sounds novel - to copy an image onto the screen, using a mouse and LineDesign (or another graphics program), try tracing the image onto acetate sheet first, then place the sheet over the QL display screen (it will stick there by static-electric attraction) and move the mouse pointer around the traced image on the acetate.

## Boats Laid Up!

Had the whole newsletter not been available, a few phrases from it would have had me more than a little puzzled. For example, "A total of 21 Squibs came together in clear blue skies" and "Squib Laying Up Supper". The Squib is a class of small yacht, subject to race meetings and end-of-season suppers for the sailors when the boats are put to bed. In several ways, Peter's use of the QL is typical of what a lot of home computer users do. If they all got results as good as he does, they would be quite happy.

Changing tack a bit, Peter mentions the Psion 3, which impresses him. My own experience of it is very limited, confined to a recent session persuading one to transfer files to a "host" computer and print to a Canon Bubblejet printer. Without having touched one before, it didn't take me long to get the hang of both operations, despite the (usual) lack of accuracy in the instructions. File transfer took the longer time to sort out, purely because the program at the host end defaulted to a Baud rate of 19,200 whereas the Series 3 offered no more than 9,600; the instructions did not mention this, as you can guess.

It is quite simple to use. Apart from the tiny keys, which make typing a pain, it has much to recommend it. Several programs can be loaded, and switching between them is easy; it is commendably nippy, and is a very convenient size for carrying around. The .DBF format of the database files in it was unacceptable to Archive, but a .LIS conversion of a database file loaded into Quill without trouble. From recent comment in magazines, it would seem that the latest Series 3 is much better than the one referred to here, and it should be possible to pick the older model up for a lower price.

## Help - Sick

### Thor

John Wakefield asks for help on something which is outside my range. Maybe a reader can help? He has a

Thor and feels that the keyboard is faulty. The symptoms are variable effects from the Alt, Ctrl and Shift keys. For instance, holding down either the Shift or Alt key for any length of time apparently causes them not to function. Ctrl-R will cause a reformat in Perfection, but a second tap of R (with the Ctrl key still held down) may cause that character to appear on the screen. On the other hand, the effect of holding the Ctrl, Shift and Alt keys down can persist some time after they are released. Is this deliberate, a software problem, a keyboard malfunction, or something quite different?

There must be many QL users who believe that all Thor activity has died out. One such user contacted me as a result of talking to the supplier Transform (yes, they are still active, but not in QL matters). The recent bout between Kasparov and Short had aroused the chess-playing enthusiasm of his mates at work, but they had battered their Psion Chess cartridges to death. He wanted to know if one of the program files could be re-copied onto his cartridge. That in itself should be a simple job (but was not); the weak link in the chain is not the copy used for playing, though, but the master cartridge which has to be there for the program to start at all, and his master would not play ball. There is a floppy disk version of Chess around, and it does not seem to need a master, but whether or not this has been doctored to work on its own, I do not know. It is unlikely that Psion still support it (for the QL, anyway).

## TF SERVICES

### MINERVA

The ULTIMATE operating system upgrade

MKII MINERVA with battery for 256 bytes ram, CRASHPROOF clock & Philips I2C bus for interfacing. Can autoboot from battery backed ram. Quick start-up.

Other features common to MKI/MKII

DEBUGGED operating system/ autoboot on reset/ Multiple Basic/ faster scheduler-graphics (10% of lighting)-string handling/ WHEN ERROR/2nd screen/ TRACE/ foreign keyboard drivers/ 'warm' fast reset/ Auto reboot after power failure. V1.97 now with built in Multibasic and split OUTPUT baud rates with Hermes.

1st upgrade: free. Otherwise £3 (+ £5 for manual if req'd) (send sac, Minerva & NEW disk/3 mds)

MKI to MKII upgrade - £30

MKI... £35 MKII(RTC)... £60

GOLD CARD (v2.24+) COMPATIBLE

### HERMES

A replacement QL co-processor for the QLs awful IPC 8049

- Do you get keyboard bounce?
- Do you find fast serial input unreliable?
- Do you want to connect a modem at 19200bps

If you can say one YES, then you need HERMES

- 19200bps RELIABLE serial input - NO QCONNECT.
- Independent input baud rates - use serial mouse & print
- Stops keyboard bounce (unwanted repeat chrs)
- Improves 'fuzzy' and 'random' sound
- Provides extra input/output lines
- Key click

Fitting is simple. Remove the QL top (8 screws) & replace the chip marked 8049 or 8749 next to mdu 1.

£25 including manual/software

### I2C Interfaces

The I2C bus was designed by Philips to simplify interfacing. Minerva MKII clock is driven by an I2C chip, & a connector allows connection of other circuits. Our external circuits will interconnect without leads. Up to 5 interfaces can be powered off the QL.

Parallel Interface Each gives 16 input/output lines. Can be used wherever logic level signals are required eg model train controllers. Can input directly to motor drivers (eg L293/298).....£25

Analogue Interface Each gives 8 analogue to digital inputs, and 2 digital/analogue outputs. For temp measurement, sound sampling etc.....£30

Data sheets (analogue/parallel I2C chips).....£2

Control software/manual (Superbasic extn).....£2

(First interface purchase includes free I5D/9D load)

### QBBS - UK's first scrolling Bulletin Board

Note the new telephone number below. TANDATA callers must add SIX zeros (000000) or wait for 3 seconds of modem tone if dialling manually.

### QL SPARES

Faulty QL board (no plug-in chips).....£9	Circuit diagrams.....£2
Keyboard membrane.....£8	1377 PA1.....£3
68008 cpu.....£8	Power supply.....£12
JM ROM.....£10	8302 ULA.....£10
8302 ULA.....£10	8301 ULA.....£10
8049 IPC.....£8	MDV ULA.....£12

Other components/sockets etc) please phone

### QL REPAIRS

Fixed price for unmodified QLs, excluding microdrives. QLs tested with Thom-EMI rig and ROM software

£27 including 6 month guarantee

Prices include post & packing (UK only). Payment by Mastercard/Visa/Access/Eurocard/cheque/postal order/PO Giro transfer (58 267 3909). MAIL ORDER ONLY - no callers without ringing first. Ring for overseas prices.



Holly Corner, Priory Road, ASCOT, Berks, SL5 8RL

Tel: 0344-890986

Fax & BBS: 0344-890987





# Open Channel

Open Channel is where you have the opportunity to voice your opinions in Sinclair QL World. Whether you want to ask for help with a technical problem, provide somebody with an answer, or just sound off about something which bothers you, write to: Open Channel, QL World, The Blue Barn, Tew Lane, Wootton, Woodstock OX7 1HA.

## Other Languages

If you are interested in programming in **Assembly**, then check out **Advanced QL Machine Code** by A. Denning, **The Sinclair Qdos Companion** by A. Pennel, **QL User** January and March 1986, 'Of Disks and Drives' by Colin Opie, and the recent **SQLW series Systematic Machine Code Programming** and **Beginners' Machine Code** by Alan Bridewell.

The strengths of **SuperBasic** have until recently kept C at bay. However, I feel the tide is changing in favour of C, especially since the release of the cheap PD C compiler (see QL World, March 1992, 'Inside the C68 compiler' by Simon Goodwin) and PD C tutorial packages. The Cport utility (see QL World Dec 1992 'Cport Review' by Shirley Butler) has much to do with the emerging popularity of C as it is able to convert SuperBasic programs into C code with the minimum effort. C has received more than its fair share of exposure in QL magazines with a number of product reviews and articles on using C - QL World September 1990, November 1990 and February 1991, 'Programming in C' by Andy Wright, and QL user 1985, 'C series' by A. Denning. If you are interested in C, then check out your local library or order copies of the following articles (your library may have these in archive) - PCW

June 1989, 'ANSI C' by N. Martin and PCW January to May 1985, 'Teach Yourself C' by L. Hampson.

Alan Wilson  
Stenhousemuir

## Good Dreams

Following Tony Rushton's very kind offer to create a **printer driver** for my Citizen Swift 24, I enclose photocopies of the escape codes. I would be extremely grateful if you could pass these along to him with the SSAE and my thanks.

Speaking of thanks, could you also thank Dilwyn Jones on my behalf for providing the **Z88** information. I would have written earlier, but I'm afraid that I've been extremely busy recently, working abroad and getting engaged among other things! It is nice to be involved in a community which is so helpful - I must confess that once I almost left the QL in favour of the Amiga, but now I'm getting a PC with QXL card once funds are available (ie not for some time!).

Lastly, would you let your readers know that my adventure game **Dreamlands**, formerly published by CGH Services, is now freely available as Charityware. I have sent Qubbesoft a copy which explains the concept in the Quill.doc file. The name of CGH still appears on the title screen but the game is now no longer a commercial one.

Jean Yves Rouffiac  
London

## Colour Stuck

In QL World Vol II.9 you published an interesting tip from Bruno Coativy on **cursor colour change**. I tried it immediately on my JM-rom QL with Gold Card 2.23. Unfortunately, it does not work, because it is not possible to poke the required colour mask into the address found by function "adresse\$" (in my case \$24A8). The contents of this address are changeless, and are set to the colour red! Having Bruno Coativy's idea in mind, I tried also to modify the original QL character definitions in Font 2, with the same negative result. The investigated ram locations of the Gold Card (\$24A8, \$A530 ... \$A770) behave in like manner to the rom. Where or what is the reason for this?

Dr. Vladislav Teply  
Prague  
Czech Republic

Stuart Honeyball writes: The address \$24A8 lies in the QL rom's address space. This ram used by the current Gold Card is write-protected to prevent programs that attempt to write to the QL rom's space from causing corruption. Earlier Gold Cards did not have this protection, and so allowed writes to this space.

## All At Sea

In Vol II issue 7 (I think the July issue) you have a review of a **naval battle**

game program, **Fleet Tactical Command**. As my employers may allow me to buy my now rather old-fashioned IBM PC 286, I would like to know where to get the serial **lead** mentioned, and how I would to connect the PC and the QL together?

Ian Marsh  
Sheffield

For all information about **FTC**, talk to Dilwyn Jones at DJC (0248 354023). However, a good source of all leads and information about them is **Tony Firshman at TF Services (0344 890986)**.

You are right about II.7 being the July issue. The issue number is the number of the month (January = 1, February = 2, and so on). We made a mistake in August and repeated II.7 on the cover, however, II.8 is correct on the Contents page. Likewise, the first issue with the Volume/issue number system, in April, has (quite correctly) II.4 on the cover, but (also correctly) "April 1993" on the Contents page, to help people pick up the new system.

We thought it would save our printers confusion, as some of the issues are rather loosely attached to their proper months at the moment. But not everyone is happy - one reader thought that II.7 was the April issue! In our defence, the photo feature in QL Scene does say "5th June". We have a few grumbles when things are late, but not normally that they are two months early!



## Looking to Help

The Royal National Institute for the Blind is seeking **volunteers familiar with personal computer technology** to help look after customers on its Electronic Newspaper service. The Electronic Newspaper is a self-funding service provided by Electronic Text Network Associates (ETNA), a joint venture company led by RNIB. Using the ETNA service, blind and partially sighted people have independent access to a daily newspaper, something which many sighted people take for granted.

The Electronic Newspaper consists of the text of The Guardian newspaper, transmitted each night via teletext Packet 31, which is received by the user on a conventional TV aerial connected to a proprietary decoded board fitted into the user's PC. The PC runs a TSR that decrypts the signal and stores the incoming files to the hard disk. When the user wishes to read the newspaper, the articles are read by synthetic speech, or on-screen by character magnification software, or by touch using a moving braille display (soft braille).

Access to computers is becoming easier for blind and partially sighted people, thanks to recent improvements in synthetic speech devices, but unless they are familiar with the technology, some users may have difficulties installing the PC decoder card and software. The user can purchase the card and software from the RNIB, but to keep down costs, the RNIB would like to find volunteers in all parts of the UK familiar with PC technology who could help our customers with installation and, where necessary, troubleshooting.

The volunteer would be asked to visit the customer when the decoder card and software are delivered, to help them fit it in their own

PC. Once the system is up and running, the volunteer may be asked from time to time to visit the customer if there are any particular difficulties. Each volunteer would only be asked to look after one or two people; travelling expenses are available.

As some of your readers may know, we already have over 3,000 valuable volunteer engineers looking after the servicing of RNIB Talking Book players. Many "servicing visits" lead to friendships, and with the Electronic Newspaper, there is the opportunity for the volunteer to demystify the PC for the user, and to learn about computer access technology for blind people.

If you would like further information about becoming an RNIB computer volunteer, please contact me at the following address:

**Prospect House,  
Prospect St., Huddersfield  
HD1 2NU. Tel. 0484  
450982. Fax: 0484 450703.**

**David Finlay-Maxwell  
PhD MIEE  
Royal National Institute  
for the Blind.**

*The RNIB are looking mainly for PC volunteers, but some QLers are PC users as well and others may know someone who would like to know about the Electronic Newspaper, or who needs similar assistance with a QL or related format.*

## New Generation

I am not sure as to the possibility of obtaining a copy of **Family Tree**, that used to be advertised in the Microdrive Exchange. I typed it from the original issue (August 1985), and have continued to use it repeatedly since, with the drawback of having to print out individual family groups! I noticed that more recent copies have an extended

family tree ability. As our family reaches back to England beyond the eighteenth century, this facility would be a great help. The name of Cooper is of course linked to the old occupation of barrel-making, so it may go back a few centuries! If Family Tree is still available, could you please let me know the price etc?

**Keith Cooper  
Morrinsville  
New Zealand**

*I have had this letter for some time and corresponded with Keith Cooper. However, as we never managed to regain our master copy of Family Tree, or make contact with author Andy Carmichael, and as there are many keen tracers of family trees still out there, I thought I would draw attention again to the newest version of Chris Boutal's **QL Genealogist 3**, available from Dilwyn Jones. The new version costs £60, but there are alternative versions at £30 and £12 as well.*

## 11.8 Or Not 11.8?

I read in QL World 11.9 that

some users have not received 11.8 - watch out as QL World 11.8 has on the cover page reported 11.7! Instead the correct number is reported in the "Contents" page. Maybe some users have been **fooled** by this printing error!

**Davide Santachiara  
Ergon Developments  
Italy**

*Despite all I've said, they're not that easily baffled! There really was a batch of 11.8 bound for Continental Europe that didn't reach its destination, and Fran has been sending replacements.*

## AArk!

More information about the **QL-generated picture** of a Kingfisher, drawn by Dave Barker of Charvelsoft, now that it has made it on to last month's cover. The image (used here with Dave's express permission) is available on a public domain disk formerly distributed by CGH Services, and now by the West Midlands Quanta sub-group disk library.

**Simon Goodwin  
West Midlands**

## Editor's Notebook

Our leader article this month is about artificial intelligence, and gives an experimental demonstration program for the QL. Dr. Steven Wilcox is an academic working in artificial intelligence, and a QL user (a sign of real intelligence), so he knows the subject from both sides.

This issue introduces the new slimmed-down QLWorld. Like most businesses that go into this euphemistically-termed operation, we are not happy about it, but, as the changes in our advertising layouts over the last few months make clear, the slow decline in the market and the pressure caused by the recession is affecting everyone who relies on the open marketplace. Oh for a bottle full of QL-using genies to do our bidding!

Our reduced space makes it more important than ever to know what you, the readers, like and don't like each month. Don't be shy - let us know.



# New Home for London Quanta

The Quanta London Subgroup is back to regular meetings at a new venue in central London.

Meetings will now be held on the second Sunday of each month, between 1pm and 6pm, at St Aloysius' Primary School, Phoenix Road, Euston, London NW1.

Phoenix Road is at the Euston end of Eversholt St, close to Euston Underground, and 10 minutes' walk from Kings Cross and Euston Square undergrounds. Several buses pass nearby, and there is car parking in the area.

The meetings are open to all interested in the QL Admission is £2 on the door to cover the cost of the Hall, and organiser **Colin Murphy (071 383 0007)** is looking for people to give talks and demonstrations on hardware or software. A few weeks' notice of talk subjects is helpful, as is notice if you intend to bring hardware, so that they can arrange enough tables.

Among regular members at the moment are Fred Toussi of Software 87, Jerry Davis of QPacer, and programmer Mark Knight.

## ALL FOR- MATS DIARY

Coming dates for the All Formats Computer Fair are:

**20 Nov** London Sandown Park Racecourse **21 Nov** Portsmouth Guildhall **27 Nov** Haydock Park Racecourse, M6 J23 **28 Brunel Centre, Temple Meads Station, Bristol** **4 Dec** Leicester De Montfort Hall, Granville Road **5 Dec** North East: Washington Leisure Centre, District 1 **11 Dec** West Midlands National Motorcycle Centre, M42 J6 **12 Dec** Leeds University Sports Centre, Calverley St. **18 Dec** Edinburgh Adam House, Chambers St. **19 Dec** Glasgow City Hall, Candleriggs **CHRISTMAS BREAK** **15 Jan** Haydock Park Racecourse, M6 J23 **16 Jan** Brunel Centre, Templemeads Station, Bristol.

The All Formats Fairs are going to many more venues now. Although you will not find QL traders at every one, they are good browsing places for multi-format users and general supplies. Check with suppliers whether they will be at a particular Fair. If you have far to travel phone **All Formats 0608 663820** to check arrangements haven't changed.

Day tickets are £4; you can get up to 50 £1-off vouchers by sending an SAE to the organisers at: Maple Leaf, Stretton-on-Fosse, Moreton-in Marsh, Gloucestershire GL56 9QX. Photocopies of these vouchers are OK. Admission is a flat £2 between 2pm and 4pm. £1-off vouchers do not apply at these times.

## Miraculous Progress

Miracle Systems came back from the **QL show** at St Joris' College, **Eindhoven**, on 30th October, well satisfied with their visit. The QXL Card software has now reached the stage where only the SBasic is to be completed, but Miracle are in a cagey mood about announcing which of their eagerly-awaited new products will be the next to appear.

Stuart Honeyball tells us: "We want to make sure that we do the new product that a lot of people want, and that we can support it properly, and we can only support a certain amount of products at a time. So we don't want to announce time scales on new products yet. In a way I wish we hadn't announced the graphics card as early as we did, but we want something really good and we are not going to compromise."

Miracle also have the services of student Benjamin Wikes to work on the SCSI interface while he is doing his sandwich course. Commenting on working standards among young engineers, Stuart said: "We find that whether people are enthusiastic or not is the most important thing", a comment that many others will agree with.

One of the challenges facing the graphics card is that much established QL software is written for the native QL graphics that the card is designed to go beyond. Programs written for the pointer environment can handle higher resolutions, but within the QL community, Stuart says, when the time comes there will need to be some rewriting of software to make use of improved graphic modes.

In line with their policy of committing their resources to current and new projects, Miracle are no longer marketing the **ED drives** which they introduced to help Gold Card users obtain reliable disk drives. The drives can now be obtained from Bill Richardson at W N Richardson (EEC) on 0753 887149.

# QL scene

## TF SERVICES BRINGS OUT I2C BUS INTERFACES

TF Services have designed two **hardware interfaces** based on the parallel and analogue I2C bus chips used by the Minerva Mk 2 rom to run its clock and battery-backed ram. Each interface contains two I2C chips, each of which can have one of eight addresses (set with DIP switches).

Power and signals are taken from the Minerva connector, and at least four interfaces can be powered without any external supply.

The TF package provides software to make reading and writing to the devices easy.

I2C compatible hardware on the general market includes phone diallers, teletext chips, and radio and TV controllers. Minerva will not allow the use of microprocessors or other devices which need dual bus masters.

More information next month. For enquiries, contact **Tony Firshman** at TF Services, Holly Corner, Priory Road, Chavey Down, Ascot, Berks SL5 8RL. See also Tony's ad. in this issue of QL World.



# SuperBasic in Action

## **Simon Goodwin unveils eccentric SuperBasic to tame the Motorola 68020, 68030 and 68040 processors.**

The new 32 bit processors in Motorola's 68000 family are upward-compatible with their eight and 16 bit predecessors, so they can run the same code. This means that Qdos users should be able to upgrade to the new chips and run their existing programs at high speed.

Much of the extra performance of the new chips comes from speed-up circuits which did not appear in the original 68008 and 68000. Some QL programs which have worked reliably for years fall foul of the new hardware, so QXL, ST and Amiga emulator users need to be able to turn features on and off. These

SuperBasic programs do that job by updating Motorola's processor control registers.

For instance, the 'code cache' stores instructions locally inside the processor. This saves time, but can cause erroneous results if you run programs that patch their own code, like Speculator, Lightning and the original Turbo Toolkit implementation of SEARCH\_MEMORY.

### **Cache Control**

The processor may carry on running the original code from the cache, ignoring the new version in memory. You can avoid this

problem in two ways - either by clearing the cache before you use the self-modifying code, or by turning it off completely. The first approach gives better performance, but the second is more reliable, especially if you do not know exactly when the cache is invalidated.

The 68020 processor, used in the Thor 20 and Amiga 1200, has a 256 byte code cache. The 68030 adds a 256 byte cache for data, and the 68040 expands both to 4K each, increasing the chance that required code or data will be available on-chip.

The 'memory management unit', or MMU, makes an appearance in top of the range MC68030 and

MC68040s, and the new LC68040. The MMU translates addresses so that memory can be re-located without the need to copy the entire contents from one place to another. Qdos has no need of a memory management unit, and program performance is improved if you turn it off.

### **Why SuperBasic?**

This month's column is a hybrid of SuperBasic and machine code. The programs are in SuperBasic, yet they access the nitty-gritty inards of Motorola's 32 bit processors. I could have written the programs in an assembler - they generate a small machine-

```

100 REMark 68040 MMU and CACHE control extensions
110 REMark SuperBasic in Action, Simon N Goodwin
120 :
130 DEFine PROCedure CACHE(d,c)
140 LOCAL x
150 IF UNSET(d) THEN d=1
160 IF UNSET(c) THEN c=1
170 x=ALCHP(20)
180 POKE_W x,20032:REMark SV mode
190 POKE_W x+2,8764:REMark Load D1
200 POKE_W x+4,-32768*(d<>0):REMark Data
210 POKE_W x+6,-32768*(c<>0):REMark Code
220 POKE_W x+8,BIN("010011100111011"):REMark MOVEC
230 POKE_W x+10,HEX("1002"):REMark Move D1 to CCR
240 POKE_W x+12,636:REMark User mode
250 POKE_W x+14,-9985
260 POKE_W x+16,HEX("7000")
270 POKE_W x+18,HEX("4E75")
280 CALL x
290 RECHP x:REMark Phew
300 END DEFine CACHE
310 :
320 DEFine PROCedure MMU_OFF
330 LOCAL x
340 x=ALCHP(20)
350 POKE_W x,20032:REMark SV mode
360 POKE_W x+2,8764:REMark Load D1
370 POKE_L x+4,0:REMark Disable MMU
380 POKE_W x+8,BIN("010011100111011"):REMark MOVEC
390 POKE_W x+10,HEX("1003"):REMark Move D1 to TCR
400 POKE_W x+12,636:REMark User mode
410 POKE_W x+14,-9985
420 POKE_W x+16,HEX("7000")
430 POKE_W x+18,HEX("4E75")
440 CALL x
450 RECHP x
460 END DEFine MMU_OFF

```



code routine as they work - but it was more sensible to develop them in SuperBasic.

The programs use the MOVEC instruction, which was introduced for the 68010 processor and later models. MOVEC allows access to special 'control' registers which do not appear in the earlier 68008 used in the QL, nor the 68000 of Amiga, ST, Thor XVI and Gold Card. Current QL assemblers do not recognise the MOVEC mnemonic, although it can be entered as a data word if you know the right code.

Recently I had the chance to try out Amiga Qdos on the EC68030 and MC68040 processors, and I wanted to use MOVEC to configure the chips. Within a few minutes I had a working SuperBasic program to generate and call the necessary code.

If you are developing a large assembler program it is best to include the MOVEC routine there, but if all you want is a way to switch the special features of your processor, there's really no need to leave SuperBasic.

The SuperBasic version is ideal for experimentation because you can edit any part and try out the change straight away. Once a skeleton routine is working you can tweak it to use any of the 680X0 control registers.

I started with a single procedure to switch the 68040 caches on and off. I only needed to change a couple of values to gain control of the Memory Management Unit, and then all the special control registers of the 68030.

## The listings

Listing one is for the 68040, including the EC and LC versions. There is no point in using the MMU\_OFF command on an EC68040 board such as

the QXL, as EC chips have no MMU.

The listing implements two commands: MMU\_OFF and C A C H E . CACHE takes two optional parameters, which control the data and code cache respectively. Lines 150 and 160 use the DIY Toolkit UNSET function to check if parameters were supplied, and set them to one otherwise. If you lack UNSET, from DIY Toolkit Volume P, you should miss out those lines, and both parameters MUST be supplied.

By default CACHE turns both caches on; CACHE 0 disables the

data cache and enables the code cache. CACHE 1,0 enables only the data cache, and cures most problems with self-modifying code. CACHE 0,0 is the slowest and most compatible option.

The POKES store code in a 20 byte buffer on the common heap. The first word, 20032, is the 680X0 code for TRAP #0, and selects Supervisor mode on a Qdos system. This is necessary because the MOVEC instruction is 'privileged' and cannot be executed in the normal 'user' mode.

The next word, 8764 in decimal, is the code to load the subsequent long word value into register D1. Lines 190 and 200 check the parameters and store -

```

100 REMark 68030 cache control commands, version 1.1, 15-11-93
110 REMark Simon N Goodwin, Andreas Rudolph, Davide Santachiara
120 :
130 DEFine PROCedure MOVECC(op)
140 LOCAL x
150 x=ALCHP(20)
160 POKE_W x,20032:REMark SV mode
170 POKE_W x+2,8764:REMark Load D1
180 POKE_L x+6,op
190 POKE_W x+8,BIN("0100111001111011"):REMark MOVEC
200 POKE_W x+10,HEX("1002"):REMark Move D1 to CCR
210 POKE_W x+12,636:REMark User mode
220 POKE_W x+14,-9985
230 POKE_W x+16,HEX("7000")
240 POKE_W x+18,HEX("4E75")
250 CALL x
260 RECHP x
270 END DEFine MOVECC
280 :
290 DEFine PROCedure DCACHE_START
300 MOVECC HEX("3911"):REMark Both on, clear data
310 END DEFine DCACHE_START
320 :
330 DEFine PROCedure CCACHE_START
340 MOVECC HEX("3119"):REMark Both on, clear code
350 END DEFine CCACHE_START
360 :
370 DEFine PROCedure CACHES_OFF
380 MOVECC HEX("2808"):REMark Everything off
390 END DEFine CACHES_OFF
400 :
410 DEFine PROCedure DCACHE_ONLY
420 MOVECC HEX("3908"):REMark Data on, clear both
430 END DEFine DCACHE_ONLY
440 :
450 DEFine PROCedure CCACHE_ONLY
460 MOVECC HEX("2819"):REMark Code on, clear both
470 END DEFine CCACHE_ONLY

```

32768 or zero in subsequent words, depending on the parameter values. The 68040 Cache Controls only use one bit, the most significant in each word. If this is set the cache is turned on. -32768 corresponds to 1000 0000 0000 0000 in binary, or 8000 in hexadecimal; it easy to see how only the top bit is used in these values.

## Motorola Op-code

The next instruction is the MOVEC op-code, which never appears in QL code, so I got the pattern from the Motorola Programmers Reference Manual, which expresses op-codes in binary. I could have con-

verted it, but there's little point in doing so, and running the risk of error, when I can type the binary directly into SuperBasic and evaluate it with the BIN function.

MOVEC is always followed by a second word, indicating the source and destination registers. In this case the most appropriate base is hexadecimal; the first four bits of the word contain the register number, 0 to 7 for D0 to D7, or 8 to F for A0 to A7.

The remaining bits select a control register; 002 signifies the Cache Control Register, and 003 the Translation Controller. There are a few other possibilities, which depend on your choice of processor; for instance 801 is the Vector Base Register, which



allows reset, interrupt and trap vectors to be re-located to an address other than zero. To find out more, consult the Motorola manual for your processor.

The next two words return the processor to user mode, once the privileged MOVEC has been done. The hexadecimal 7000 opcode sets register D0 to zero. This is one of the few 68000 instructions that's easy to assemble by hand. The seven indicates MOVEQ, the second hex digit is the register number times two, so 74 would move a value to D2. The last byte is the value to be moved, so 7608 is the code for MOVEQ #8,D3. Easy!

The last instruction is 4E75 in hex, "Nu" in Ascii, and RTS in mnemonic assembly code. It marks the end of the code and returns control to the caller - SuperBasic's CALL, in this case. The final step is to deallocate the buffer with

RECHP.

MMU\_OFF needs no parameters. The bulk of the code resembles that for CACHE. A sequence of 20 bytes is stored and then called, but in this case the code clears TC, the Translation Control register.

## 68030 Range

Listing two is for the 68030 range, tested on an Amiga 4000/EC 030. The 030 uses a different control register format, with extra bits to clear all or part of either cache, so you can't just use the 68040 version.

The listing implements five commands: CACHE\_OFF, which works like CACHE 0,0 on the 68040; DCACHE\_ONLY, like CACHE 1,0, and CCACHE\_ONLY, which turns on the code cache but not the data one.

The remaining commands leave both caches

on, but clear one or other so new memory contents are recognised. CCACHE\_START clears the code cache, while DCACHE\_START does the same for data. All five procedures call MOVECC, which sets up the bulk of the MOVEC code and stuffs a parameter into the low word of D1.

## Conversions

The program uses base-conversion functions HEX and BIN from Toolkit 2. The HEX function also appears in Mark J Swift's PD Toolkit. If your super-QL lacks these functions you can replace the values with their decimal equivalents, but this will make them more difficult to change, particularly in the case of BIN, where each binary digit has a special significance to the processor.

The generated code is

stored in a temporary area of memory allocated on the common heap. The program uses the Toolkit 2 function ALCHP to allocate memory, and RECHP to discard it after use. If you lack Toolkit 2 you may use either the eponymous PD Toolkit functions, or ALLOCATION and DEALLOCATE from Turbo Toolkit, or RESERVE and DISCARD from DIY Toolkit Volume H.

## Resume

This is not a typical SuperBasic in Action project, but it does show the sort of low-level access that is possible with POKE and CALL. Next month I shall be back with more conventional SuperBasic programming. In the meantime I hope 32 bit Qdos users find these little procedures useful, and welcome suggestions for future columns.

# JOCHEN MERZ SOFTWARE

Im stillen Winkel 12 . 47169 Duisburg . Germany . Tel and Fax: 0203 501274 . Mailbox: 0203 591706

### typeset93-ESC/P2 NEW!

Dedicated text87 printer drivers for all EPSON printers with ESC/P2 including the new Stylus 800. Supports the scalable fonts, special characters and block graphics for drawing borders etc. DM 69,90

### QMAKE NEW!

Pointer-driven Make program for the GST/Quanta Assembler package. Very comfortable, with many options including library-building! DM 44,90

### NEW!

**The Oracle** - an ancient tactic-puzzler. You don't have to be fast but clever! Fill different tiles into a field, but there are various rules how to do it. If you can place all the tiles you get a bonus - try to obey special rules. Every game is different, you'll never be bored. Features: high-score table, hints for the next move etc. DM 39,90

**QD Version 5** - Very comfortable Editor for the Pointer Environment now with Help System! [V5.16]. DM 125,- Upgrade from QD V4 (return master disk) DM 30,-

### HyperHELP for SuperBASIC

Instant help in SuperBASIC! It displays a list of SuperBASIC Procedures, functions and keywords currently existing in your machine. Simply click on the name and you get extensive help (English or German). The help files can be updated with any editor, QD, Quill, Text87 so that you can update and add remarks whenever you want. DM 49,-

**QMenu - the Menu Extension.** QMenu is an interface with pre-defined menus (e.g. multi-column file-select, simple-choice boxes, select from lists). The menus can be used from SuperBASIC, machine code and other languages. New character-select menu. [V4.16] DM 39,90 Update with new manual DM 16,-

### BlackKnight - the first mouse-driven Chess program!

Yes, here it is - chess for the Pointer Environment! Moveable, runs on any display resolution! 5000 moves opening library, demo made, setup, change sides, load and save game, hint, scrollable move-history etc. 10 Levels of difficulty, giving the computer between 5 seconds to 1 hour time to think.

BlackKnight even thinks when its your turn. DM 119,90 NEW!

### The Lonely Joker Version 2!

Now six different games in one! Another three, even more complex patiences have been added to the existing three: Spine, Crapaud & Four-in-a-Hand (two of them work with two packs of cards). More options (different patterns for the reverse side of the cards), high-scores, moveable window for hi-res screens etc. DM 59,- (double fun-half the price) DM 29,- NEW!

**QSpread** - the only Spreadsheet program for the Pointer Environment. Completely mouse and/or keyboard-controllable! Maximum 32768 cells, where, for example, 16000 cells need about 400kBytes. Macro functions. Easy, quick and comfortable block handling and entry. Many additional functions, which belong to today's standard-equipment: Help, Button, use of the Scrap. The current version adds missing features and more, e.g. cell-reference-adjustment, echo, IF and comparisons, toolbar, time & date functions. [V1.22] QSpread with comprehensive manual DM 169,- Update with new manual (useful if old version is <1.20) DM 16,- QSpread Demo Version - send 3 International Reply Coupons if ordered alone, otherwise enclose a spare disc with an order or update and it is free!

### QL Hardware & Spares

FLP/RAM Level 2 device drivers for SuperQBoard  
FLP/RAM Level 2 device drivers for TrumpCard  
SER Mouse software driver  
SER Mouse Package (mouse, adaptor & driver)  
ZX8301  
ZX8302  
QL Keyboard membrane DM 28,-  
branes  
EPROM 27128  
SIMM 256k x 8

DM 56,-  
DM 56,-  
DM 40,-  
DM 87,-  
DM 19,90  
DM 17,90  
2 mem-  
DM 50,-  
DM 5,-  
DM 3,50

### QL-Emulator for ATARI

QVME - High-Res QL-Emulator for ATARI  
Mega STE and ATARI TT DM 695,  
E-Init Emulator Software [E34] DM 49,-  
E-Init Update with new manual DM 16,-  
QL-Emulator Brochure DM 13,-  
Medium for SyQuest 44MB DM 139,

### Applications

QDOS Reference Manual DM 89,9  
FIFI DM 49,9  
QPTR DM 92,-  
QSUP DM 79,9  
QLQ DM 69,9  
Thing & EPROM Manager DM 61,5  
EasyPTR Part 1 DM 89,-  
Part 2 DI DM 49,-  
49,- Part 3 DM 149,  
DataDesign V3 DM 80,-  
SYSTEM DM 150,  
QDesign II DM 60,-  
VecEdit

### Games

Diamonds DM 35,9  
BrainSmasher DM 45,9  
Arcanoid DM 35,9  
Firebirds DM 35,9  
SuperGamesPack DM 90,-  
G'Shang DM 45,9

**Merry Christmas & Happy New Year !**

Postage and package (Europe) DM 15,- (if total value of goods is up to DM 50,- then only DM 9,-)  
(Overseas) as before, but add DM 8,- per additional item. All prices incl. 15% VAT. (can be deducted for orders from non-EEC countries). E&IO. Cheques in DM, £s, Eurocheques and Credit Cards accepted.





# QLScene

## French AGM

French QL Group QL Contact France are holding their Annual General Meeting in Paris on Sunday 5th December. Contact Jean Louis Dianoux, 22 Ave. Lenine, 93230 Romaineville, France.

## Starting With The Prolog

Grange Technology of Didcot have introduced a new high performance implementation of Prolog for the QL **GT-Prolog** can be used on any QL with 512K or more of memory. Combining speed with low memory overhead, the program provides an interactive development environment supporting incremental compilation and debugging of Prolog applications.

GT-Prolog is a full implementation of the Edinburgh dialect used widely in industry and education, and as the basic standard referred to by the British Standards Institute and International Standards Organisation. This implementation is also highly compatible with other major Prolog systems, giving immediate access to existing application code, documentation and training material.

GT-Prolog features a full range of data types and gives transparent access to optimisation techniques including In-line Arithmetic Evaluation and automatic Garbage Collection of code and data. It is also one of the first implementations to support a fully logical Prolog database. It provides explicit control of memory allocation, allowing programs up to 16 megabytes. Execution speed measured using Naive Reverse exceeds 5K LIPS (logical instructions per second) on a 16 MHz Gold Card.

The associated Workbench gives a powerful integrated toolset for code development, with a fast incremental compiler, a multi-port debugger on the Byrd model and a source code model, all employing a menu/dialogue user interface. The Workbench is implemented in Prolog using the same 130+ family of library predicates which provide user's programs with full access to Qdos facilities.

Prolog has gained widespread use in symbolic processing and artificial intelligence, and increasingly in less specialised applications. The GT-Prolog package includes software on disk, and an 150-page A% User Guide and Manual. The single-user price is £89.85 inclusive of VAT and carriage.

For more information contact **Graham Thwaites, Grange Technology Ltd.,** Rosebank, Stream Road, Upton, Didcot, Oxon OX11 9JG. Tel. 0235 851818.

## Ergon make prices simpler

Ergon Developments in Italy have altered their pricings - to include all post and packing and bank charges. This will be gladly received by users who had to keep adding up the noughts on the Lire prices! As well as the arithmetic boon, all Ergon's manuals have been revised and extended, and have tutorials with many step-by-step examples, and are also laser printed for extra clarity. Existing Ergon customers who would like the revised manuals should contact Ergon for information.

Ergon's **DEA Intelligent Disassembler** is now into version +3, with interactive machine code decoding and a new 52-page manual. The cost is £28. **MasterBasic** 1.43 costs £22, Q-Library Manager 2.11 is £18, **Ergon Floppy Disk Utilities** is £18 (this is version 1.13, with disk utilities for Gold Card High Density and Extra Density disk users, allowing a format of 5% extra capacity, +160 kilobytes on ED disks); **Open World** is £18 and **Music Manager** is £12.

On the **Spectrum Emulator** front, both ZM/hT (which runs at full ZX speed on a Gold Card QL, as well as a reasonable speed on standard QLs) and ZM/128 now have 3-channel sound-chip emulation through the QL beeper. ZM/hT is now up to +1 while ZN/2 and ZM/3 have been upgraded to /hT supervisor level, +3. ZM/ACcessories now also include Tapefile to Speculator conversion in both directions, with a new 90+ page manual. Emulator prices are: Bargain package (ZM/hT, ZM/128 public domain versions + ZM/AC manual, £20; Budget package - as previously, plus ZM/2 and ZM/3 £30; Professional package - budget package plus ZM/hT £42. ZM/128 can be added to any of these for an extra £10.

For the updated **Ergon Demo Disk**, with PD or demo versions of most of these programs, either send 6 IRCs (international reply coupons) to Ergon, or apply to your nearest Public Domain dealer (Qubbesoft or SJPD in the UK). There is also a special Spectrum emulator PC disk with ZM/hT and ZM/128 PD and some games and shareware.

Contact Ergon at: **Davide Santachiara, Via Emilio De Marchi 2, 42100 Reggio Emilia, Italy.** Tel. (from UK) 010 39 522 70409. Please make payments with orders out to Davide Santachiara.

**SQLUG** are holding a workshop in Edinburgh on Saturday 12 February 1994. For details get in touch with Alan Pemberton, 68 Lingerwood Road, Newtongrange, Nidlotian EH22 4QQ.



# THE NEW USER GUIDE

## KEYWORD INDEX

SECTION  
TWENTY  
NINE

*In the last part of the Keyword Index, Mike Lloyd goes through WHENs, WIDTHs and WINDOWs*

WHEN <expression>  
<commands>  
CONTINUE  
ENDWHEN

WHEN <expression> :  
<commands>: CONTINUE

WHEN var  
[Minerva]

### EVENT-DRIVEN PROGRAMMING STRUCTURE

<expression>	Can be any expression involving a single variable that can evaluate to true
<commands>	Can be one or more SuperBasic statements
var	Can be any variable (its value is ignored)

The WHEN keyword appeared on a quite early version of the QL's rom, but was never adequately implemented until Minerva came along. While late Sinclair roms can just about make sense of a WHEN structure it is foolhardy to use it as it is very unreliable. However, Minerva users can make full use of the structure to add an exciting new level of sophistication to their programs.

A WHEN structure is similar to an IF structure, but instead of evaluating the expression immediately and taking the appropriate action, the SuperBasic interpreter will remember the WHEN expression and as soon as it becomes true it will nip back and execute the commands contained in the WHEN structure. This behaviour can be changed by executing another WHEN command referencing the same variable, or by issuing a "bare" WHEN command comprising WHEN followed by a variable. WHEN statements can appear anywhere in a program but only take effect after the interpreter has read the structure. Use CONTINUE to return to the line following the one that triggered the call to the WHEN code.

You can include several simultaneous WHENs in a program provided each refers to a different variable in the logical expression or the variable used is declared LOCAL

WHEN ERROR  
<commands>  
ENDWHEN  
[Minerva, some late  
Sinclair roms]

### EVENT DRIVEN PROGRAMMING STRUCTURE

<commands>	One or more SuperBasic statements.
------------	------------------------------------



WHEN ERROR is very similar to the WHEN structure described above, but the statements in the structure are executed only when an error occurs. WHEN ERROR can be used in conjunction with the SuperBasic keywords RETRY and CONTINUE in an attempt to circumvent the error. If you leave a WHEN ERROR block using GOTO or a procedure/function call the interpreter still imagines that it is inside the error handler, which is very likely not what you want. For a full explanation of error trapping see the New User Guide's forthcoming Concepts section.

WHEN\_ERROR flag  
<commands>  
RETRY  
ENDWHEN

WHEN\_ERROR flag  
CONTINUE  
ENDWHEN  
[Turbo Toolkit]

#### EVENT DRIVEN PROGRAMMING STRUCTURE

flag	Either 0 or 1, indicating the error level being defined
<commands>	One or more SuperBasic statements

Turbo Toolkit's WHEN\_ERROR (note the significant underscore) has the enormous advantage of working perfectly on every QL rom variant. It also offers two levels of error-trapping. If an error occurs in a WHEN\_ERROR 1 block, or if the end of such a block is reached without a RETRY command occurring, then the most recent WHEN\_ERROR 0 structure will be executed. Error processing can be cancelled only by writing a WHEN\_ERROR construct that contains just one command: CONTINUE.

In compiled code, no line numbers and statement locations can be assumed to exist, so the RETRY command does not work in the conventional way. Instead, program execution resumes at the line following the last RETRY\_HERE statement to be encountered. RETRY\_HERE should be strategically placed at the start of several procedures so that reprocessing can commence from logical points. Note that the Turbo compiler has no concept of "roll-back": it will not restore matters to exactly how they were when the RETRY\_HERE command was first read.

WIDTH #chan, columns

#### OUTPUT CONFIGURATION COMMAND

#chan (Optional)	A channel opened to a non-console device
columns	The maximum number of columns to print per line

The WIDTH command, on early Sinclair roms at least, is not particularly reliable. It is supposed to limit the line width of output directed to printers to whatever is stated in the WIDTH parameter. In practice, it is much easier to send a printer control code.

WINDOW #chan, width,  
height, xpos, ypos  
[SuperBasic]  
WMON mode  
[Super Toolkit II]  
WTV mode  
[Super Toolkit II]

#### SCREEN HANDLING PROCEDURES

#chan (Optional)	A valid console channel
width	The width of the window in pixels (assuming high resolution)
height	The height of the window in pixels
xpos	The horizontal location of the top left corner of the window
ypos	The vertical location of the top left corner of the window
	mode 0, 4 or 512 for high resolution, 8 or 256 for low resolution

The WINDOW command allows you to reconfigure screen windows without redefining them with the OPEN command. Regardless of the screen mode all co-ordinates and sizes assume that there are 512 pixel columns on the screen. In low resolution mode, of course, only even values are significant.

To ease the task of returning windows to their default configuration, Super Toolkit II contains two handy commands: WMON restores the initial layout of high resolution screens and WTV restores the layout of low resolution screens. They do not, however, change the current resolution. This is managed by the parameter that follows each command.



# The **NEW USER GUIDE**

## *Concepts Section*

### **Section Thirty**

**I**t is possible to use a computer without knowing much about how it works, just as you can drive a car without understanding its engine. However, to get the most from a computer, as with a car, it is highly desirable to know something about its internal workings. Otherwise you may find yourself doing the equivalent of driving slowly uphill in top gear and wondering why your computer is letting you down.

The Concepts section is your guide to the theory of computing with the Sinclair QL. This Concepts section reaches further than the original Sinclair guide and tackles topics in greater depth.

Technical terms are avoided wherever possible and the unavoidable ones are carefully explained. No prior knowledge is assumed. Whereas the original user guide dealt with its concepts in alphabetical order, the New User Guide's Concepts section is arranged so that hardware issues are covered first, followed by Qdos and SuperBasic topics. The first two topics provide an overview of the entire system from hardware to software.

### **Towards a unified theory**

Why is computing so difficult to understand? Some people have a natural gift for seeing life from a computer's perspective, but others suffer from extreme forms of technophobia. Most of us think we will know enough to cope just as soon as we've mastered a little bit more detail. Computing is difficult for a number of reasons. Firstly, it is a very immature technology: we frequently haven't found the best way of doing things yet. Secondly, development in computing science is the fastest known to human society, with technologies continually leapfrogging each other. Thirdly, computers are a new class of brain-assisting machines; all other machines are there in some capacity to assist human musclepower. Fourthly, computers are chameleons whose purpose and appearance changes with each program that is loaded. Finally, and most confusingly, there is nothing like computing for contradictions, illusions and virtuality. Nothing in computing is what it seems.

It was once observed that any sufficiently advanced technology is indistinguishable from magic. To medieval people, things like railway engines and aircraft would fall into this category. For early Victorians, transistors, calculators and television would have been magic. For us it is fractal geometry, the unified theory and true artificial intelligence. Magic is arbitrary, fathomless and unpredictable. Technology, on the other hand, is predictable, understandable and within the control of those who understand it. The difference between magic and technology is knowledge.

### **Concepts for knowledge**

Sherlock Holmes thought that the brain had a finite capacity for knowledge. When he was told that the Earth revolves around the Sun, he dismissed the information as valueless and tried to forget it in order to leave room in his mind for something more useful. Whether the brain's capacity is fixed or not, it often seems as though it is, so what chance do we have of keeping up with the fastest-growing body of knowledge ever known?

The answer is concepts. Information can be compressed into concepts that can be understood and remembered even though the fine detail might be missing. In this way, Stephen Hawking can explain the mechanics of the universe without resorting to blackboards full of equations, I can understand the concept of fuel injection without being in the least able to design a fuel injection system, and you can understand what makes your computer work without a doctorate in computing science or electronic engineering.

Concepts are particularly valuable in computing because computer systems "exist" at many different levels and mutate alarmingly according to the perspective from which they are viewed. From the electron's point of view, computers are enormous quantities of tiny circuits that are either switched on or switched off. From the hardware designer's point of view, computers tend to be chips linked by wire pathways (known as buses): how precisely the internals of each chip go about their business is not particularly relevant. From the operating system's point of view a computer system is a central processing unit, a working memory area and links to



devices, or peripherals, between which information in the form of integer numbers can flow. From the programmer's perspective, a computer is a repository for information, often in the form of text or real numbers, and for the instructions that process the information. From the user's viewpoint, the computer is one minute a games console and the next a word processor. None of these perspectives tells the whole story, but each depends on its predecessor, like the storeys of a house of cards.

## Something out of nothing

Like Sherlock Holmes, someone concentrating on one of the levels tends to "hide" much of the levels above and beneath, so when a user views text on a computer screen all trace of the hardware layer has disappeared. There is no text on the screen, simply dots of light. There is no programming language that people can read, but only impenetrable forests of numbers. There are in fact no decimal numbers, but only binary digits. There are, of course, no binary digits, but only pulses of electrical charge.

If you turn off a car's engine, it is still demonstrably a car. You can service the engine, sit in the seats, polish the paintwork. You can release the handbrake and push it around. Switch off a computer and it's a doorstop. The computer's marvellous edifice of abstraction upon abstraction collapses back into its component parts, like a soap bubble, literally as if it had never been. Computers never learn anything, they never get better at anything, and they never profit from experience. Even computer programs that log previous behaviour never get any better at interpreting it.

Computers are real-life equivalents of Doctor Who's Tardis: their external and internal dimensions are entirely contradictory. No other machine ever invented has this capacity to exist at several different levels and take on such different appearances at each level. In the computer, things that appear to be significantly different physically, such as a microdrive cartridge and a random access chip, might at some level become completely indistinguishable. Things that are physically similar, such as floppy diskettes, might be readable on one type of computer and completely unreadable by another. Something made up of many things, such as the QL's random access memory which is spread across several chips, might actually be one (supposedly) seamless whole. It all depends on the level at which you look at it.

## Appearance is misleading

Just as with the Tardis, size is almost always misleading. Physically, the largest chip in a QL is the central processing unit, but its internal capacity is dwarfed by the storage ability of the much smaller random access memory chips. Double density, high

density and extra density diskettes all share exactly the same dimensions, but each has double the storage capacity of its predecessor. A physically large computer, such as an IBM 286, might have less storage capacity and operate more slowly than the smaller QL. Even when its "true" internal capacity is measured, an IBM PC often uses its memory in such a way that a QL with much less memory might actually have room to store more information.

To cap it all, appearances can be doubly deceptive. A computer might look like a QL on the outside, but it might be running a program (called an emulator) that makes it think that it is a Sinclair Spectrum on the inside. You can fool an Atari ST into thinking that it is a QL and even buy an expansion card for an IBM PC that, to all intents and purposes, it really is a QL. To return the compliment, a QL can be convinced that it is an IBM PC.

It is also difficult to determine the extent of a computer system. Does it include peripherals, such as printers and monitor screens, or not? If a floppy disk drive is a peripheral because it is in a separate box and linked by cable, is a microdrive a peripheral even though it is physically inside the computer casing? The QL has a keyboard on top of its casing, but can also have an external keyboard added: are they both peripherals? Most people understand keyboards, disk drives, monitors and so on to be "devices" and thus peripheral to the computer proper even if they are integral to the computer case, but having gone down that route where do we stop? Is random access memory a peripheral or part of the computer? Is the central processing unit "a computer on a chip"? Is there a better answer than "it depends"?

## Working through the layers

Even when we tie down the computer so that it is a QL on the outside and a QL on the inside, it can be a word processor, a number-cruncher, a golf game or a chess player - or all these things at once. Computers are electronic chameleons employing layers of abstraction to turn millions of tiny, simple electronic circuits capable only of being "on" or "off" into, for instance, sufficient calculating power to outwit all but the very top human players at chess. And then, as soon as they are turned off, they forget completely that they have ever done it.

To make sense of these contradictions we must take a journey through the layers, beginning with the physical components and ending with the high-level programs, from what the Americans call "close to the iron" to what is becoming increasingly familiar as "virtual reality".

Next month Mike Lloyd starts to analyse the rock bottom concepts of computing.



## **Simon Goodwin tackles serial mouse interrupts with 68000 assembly language.**

**T**his issue completes the source listing for the DIY Toolkit mouse extensions, and includes full assembled code to suit PC Serial mice with three buttons.

I shall explain the exact format of information sent by the mouse, and show how it is converted into standard pixel co-ordinates and button signals which can easily be read from SuperBasic.

### **Assembly Code**

**Listing One** completes version 1.6 of the DIY mouse handler, with code for the remaining SuperBasic commands and the interrupt driven handlers that read signals from the mouse and store them for your program to read.

The first section of the assembly listing implements four new SuperBasic commands, discussed

last month. The code is very straight-forward as they all work the same way, reading two integer parameters with the CAGTUN vector and storing them among the mouse driver's own variables.

Qdos calls its polled interrupt handlers fifty times a second, unless interrupts are temporarily prevented by network, disk or microdrive accesses which demand the processor's full attention.

The DIY Mouse handler is labelled SERVER, and comes after the code for the commands. It starts by checking to see if the serial port is in use. If not, it returns at once.

Otherwise it reads bytes from the serial input queue until there are no more left. After processing each byte the handler returns to NEXT\_BYTE to pick up another. Address registers A1, A2 and A4 are set outside the loop and not modified thereafter, to save time when processing several bytes at once.

### **Variables**

Qdos sets register A3 to point eight bytes before the interrupt linkage when any interrupt handler is called. The LEAL instruction in my code points A4 at the variables used by the mouse driver, so later lines can find them easily from their offsets.

The **Table** shows the variables and their meanings. Variables 10 to 13 are only used in the version of the handler with keyboard emulation. Each variable is a word, normally an integer in the range 0 to 32,767.

The serial ID occupies two words; the first is the channel

tag, and the second is the number in the Qdos channel table. Both are zero if the channel was already open when the PTR\_ON command was issued.

The eighth word holds a byte value, corresponding to the SYNCH% function, which cycles from 0 to 3 or 5 as each byte of the mouse message is received and checked. The other byte in this word is only used in the Microsoft two-button version, when it stores the first byte of each message so that it can be used to extend the range of the X and Y delta bytes.

You can read some of these variables directly from SuperBasic extensions. For instance PTR\_X% and PTR\_Y% read LATEST\_X and LATEST\_Y respectively, and BUTTON% reads BUTTON\_BITS.

The next section of the code differs wildly depending on the type of mouse. Mouse Systems and Microsoft cannot agree on whether rising co-ordinates indicate movement up or down, so the three-button code SUBTRACTS incoming moves from the current position, while the two button variant ADDS them.

The code variations ensure that the result of a move is the same to SuperBasic, regardless of the model. If you represent the mouse position with a pointer drawn with pixel co-ordinates on the screen, mouse movements shift the pointer the same way - as long as you have the buttons pointing away from you and towards the screen.

Thankfully both firms agreed that values should ascend from left to right, but even that may confuse Arabic and Hebrew writers, who scan the opposite way - as digits are still written in arithmetic. Standards are tricky things.

The two button mouse moves the button bits to the right place with a shift and an OR instruction, but this

would be more complicated for three buttons, especially as they do not come in the standard order we want to hold them in BUTTON\_BITS.

The program uses a small table, labelled BUTTON\_CODE, to translate the three incoming bits into a BUTTON% value. The bits are inverted as well as shuffled, as Mouse Systems set a bit to one UNLESS the corresponding button is pressed.

Tables are wonderful things, and can often do several jobs at once in programming. They are also easier to test, modify and prove correct than algorithms. If your program needs to transform one short pattern of bits into another, it's probably best, and almost always fastest, to use a table.

### **Loose Buttons**

I should explain that the buttons are numbered the same way regardless of how many there are. The bits are in different places depending on the type of mouse, but the DIY Toolkit mouse driver hides such differences from the programmer.

As far as SuperBasic is concerned, the leftmost button is button one and the right button is button two. If both buttons are pressed, BUTTON%(0) returns three. The middle button only appears on a Mouse Systems mouse. Contrary to my last article, this is button three, ignored in Microsoft mouse mode. If this button is pressed it adds four to the result of the BUTTON%(0) function.

You can use this function to derive a QL Mode 8 colour, between zero and seven, from the combination of buttons held down as the mouse moves. This soon gets tiring if you're trying to draw a multi-coloured picture, so it's better to show the colours in a reserved area of screen and let the user select them by pointing



and clicking.

Most mouse-driven operations need only one button, which saves the user the trouble of working out which to press; indeed, Apple Mac users get no choice in this regard.

The commands were discussed last month, so I shall not repeat their details here. Arcwind can supply the back issue if you missed it; you may also find the previous one (Volume 2 number 9) interesting as it includes three SuperBasic prototypes for the mouse drivers.

## **Protocol**

Further investigation reveals that the three-button mouse returns five bytes of information at a time. The first is easily recognised as only three bits ever change - these are the button signals. The MOUSE\_KEYS program in issue 9 did not count the bytes, but reset itself to the start of the sequence whenever it found a plausible button byte, and looked for X and Y changes thereafter.

It turns out that the three button mouse sends four bytes of directional information for every one byte of button data. The SuperBasic version works fine - at least on Hermes - as it treats bytes as X and Y moves alternately till it runs across a new button code. Button codes look like extremely big negative moves, so they are easily distinguished.

## **Resynchronisation**

The machine-code version can go faster by reading the serial input queue directly, but it can still get out of step if Sinclair's old IPC loses bytes on their way from the mouse to Qdos. This is particularly likely if the QL is busy with a device and cannot read serial bytes for a while. They pile up in the 8049 co-processor, and soon overflow.

The DIY mouse handlers check the format of the first byte and if it is implausible they discard it and wait for a valid one. This means that the handler re-synchronises itself at the start of the next complete message.

Both handlers jump to the code labelled CLEAR\_SYNC when looking for the start of a new message. The two button version can also verify vertical and horizontal moves; if it finds a button signal when it expects a move, the RESYNCH routine sets SYNCH% back to one, the start of the message, and gets on with processing the button code. The bit prefix is the same for both X and Y bytes, so there's still a risk that the driver will misinterpret a Y byte as an X move if you're unlucky and only the second byte is lost.

It's not perfect, but it works most of the time; if you want perfection, get Hermes, or a real serial port rather than Sinclair's version, which is generally fine for output, but not so good at input.

## **Re-assembly**

Listing One was developed and tested with HiSoft's Devpac 2 assembler. Symbols like BUTTONS were set up at the start of the listing, and control conditional code for the two and three button versions. Don't try to assemble this listing on its own.

Append the code in Listing One to last month's source, and load the result into your own assembler if you want to modify or experiment with the full source.

It ends with the serial device name, which depends on the chosen port, and the table of command and function names. These come after the code to make it easier to change them without re-assembly, but it is wise to re-assemble the entire source if you want to change the length of any names, and you can't change the serial port with

Most mouse-driven operations need only one button, which saves the user the trouble of working out which to press; indeed, Apple Mac users get no choice in this regard.

The DIY mouse handlers check the format of the first byte and if it is implausible they discard it and wait for a valid one. This means that the handler re-synchronises itself at the start of the next complete message.

The commands were discussed last month, so I shall not repeat their details here. Arcwind can supply the back issue if you missed it; you may also find the previous one (Volume 2 number 9) interesting as it includes three SuperBasic prototypes for the mouse drivers.

Both handlers jump to the code labelled `CLEAR_SYNC` when looking for the start of a new message. The two button version can also verify vertical and horizontal moves; if it finds a button signal when it expects a move, the `RESYNCH` routine sets `SYNCH%` back to one,

## Protocol

Further investigation reveals that the three-button mouse returns five bytes of information at a time. The first is easily recognised as only three bits ever change - these are the button signals. The MOUSE\_KEYS program in issue 9 did not count the bytes, but reset itself to the start of the sequence whenever it found a plausible button byte, and looked for X and Y changes thereafter.

## Re-assembly

bytes of directional information for every one byte of button data. The SuperBasic version works fine - at least on Hermes - as it treats bytes as X and Y moves alternately till it runs across a new button code. Button codes look like extremely big negative moves, so they are easily distinguished.

**Listing One** was developed and tested with HiSoft's Devpac 2 assembler. Symbols like `BUTTONS` were set up at the start of the listing, and control conditional code for the two and three button versions. Don't try to assemble this listing on its own.

Append the code in

# Resynchronis- -ation

The machine-code version can go faster by reading the serial input queue directly, but it can still get out of step if Sinclair's old IPC loses bytes on their way from the mouse to Qdos. This is particularly likely if the QL is busy with a device and cannot read serial bytes for a while. They pile up in the 8049 co-processor, and soon overflow.







start the mouse handler.

Notice that the three-button code file is shorter than the Microsoft serial mouse version. There are more buttons, and Mouse Systems messages are longer, but there is no need to shuffle bits between data bytes to get the co-ordinate updates.

If you don't like the names of the commands and functions you can patch or ALIAS them, as I have explained in past columns. If you need to remove them from the name table, with a routine like FORGET from DIY Toolkit Volume B, make sure you turn them off first with PTR\_OFF, or the interrupt will be left running and you will no longer be able to stop it.

The processor time used by the driver is almost unmeasurable when the mouse is still, but it can slow the machine down by a few per cent while the mouse is moving fast and sending many bytes per second.

A set of DIY Toolkit mouse drivers is included in the latest volume on disk, available by post from former Quanta editor Dr. Bill Fuggle. Volume I includes the SuperBasic prototypes, wiring details, example programs, assembler source and binary code for "Mouse Systems PC" and "Microsoft" serial mouse drivers to suit either serial port. These come in two versions, one of which sends key signals so it can be used with programs like Quill, Abacus, Turbo and editors which normally only work with cursor keys.

DIY Toolkit volumes cost three pounds each on disk or microdrive cartridge, and come with printed documentation if you order two or more. Twenty four volumes are available from DIY Toolkit, 86 Lordwood Road, Harborne, Birmingham B17 9BY. Please make cheques payable to DIY Toolkit, and send a stamped self-addressed envelope if you would like further details.

## Next Issue

Next issue features the final part of this project, with add-on code to let the mouse handler generate key-presses, a few extra features, and a checklist for reliable pointing.

### Table: Internal Mouse Variables

Name	Word Number	Description
Latest_X	0	Current X co-ordinate
Latest_Y	1	Current Y co-ordinate
Limit_X	2	Right margin limit
Limit_Y	3	Top margin limit
Step_X	4	Counts per horizontal move
Step_Y	5	Counts per vertical move
Button_bits	6	Bits shadow each button
Synchro	7	Input byte number 1 to 3/5
Initial	7	Microsoft initial byte
Serial_ID	8+9	Zero or serial tag & channel
Drift_X	10	Accumulated X drift
Drift_Y	11	Accumulated Y drift for keys
Key_flag	12	Set to request key queueing
Wrap_flag	13	Set to allow wrap at edges

### Three button SERIAL MOUSE SuperBASIC DRIVER LOADER

```

100 REMark Sinclair QL World HEX LOADER v 3
110 REMark by Marcus Jeffery & Simon N Goodwin
120 :
130 CLS: RESTORE : READ space: start=RESPr(space)
140 PRINT "Loading Hex..." : HEX_LOAD start
150 INPUT "Save to file...":f$
160 SBYTES f$,start,byte : STOP
170 :
180 DEFine FuNction DECIMAL(x)
190 RETURN CODE(h$(x))-48-7*(h$(x)>"9")
200 END DEFine DECIMAL
210 :
220 DEFine PROCEDURE HEX_LOAD(start)
230 byte = 0 : checksum = 0
240 REPEAT load_hex_digits
250 READ h$
260 IF h$="*" : EXIT load_hex_digits
270 IF LEN(h$) MOD 2
280 PRINT"Odd number of hex digits in: ";h$
290 STOP
300 END IF
310 FOR b = 1 TO LEN(h$) STEP 2
320 hb = DECIMAL(b) : lb = DECIMAL(b+1)
330 IF hb<0 OR hb>15 OR lb<0 OR lb>15
340 PRINT"illegal hex digit in: ";h$ : STOP
350 END IF
360 POKE start+byte,16*hb+lb
370 checksum = checksum + 16*hb + lb
380 byte = byte + 1
390 END FOR b
400 END REPEAT load_hex_digits
410 READ check
420 IF check <> checksum
430 PRINT "Checksum incorrect. Recheck data.":STOP
440 END IF
450 PRINT "Checksum correct, data entered at: ";start
460 END DEFine HEX_LOAD
470 :

580 REMark Space requirements for the machine code
590 DATA 588
600 :
610 DATA "43FA01DA34780110","4ED2616267507220"
620 DATA "740070184E414A80","665645FA012C214A"
630 DATA "000421470008217C","01FF00FF0010217C"
640 DATA "0006000A00142C08","70004E414AA8009C"
650 DATA "661641FA01907200","760170014E424A80"
660 DATA "661022462348001C","2046701C4E417000"
670 DATA "4E7520462E007019","4E4120074E75617A"
680 DATA "4E7561FA6B70202C","0010670620407002"
690 DATA "4E4241ECFF4701D","4E41701960CE3478"
700 DATA "01124E92667C5343","66763631E800614A"
710 DATA "382C000C67364A43","6732B67C00036260"
720 DATA "5343070466267800","6022612E7800182C"
730 DATA "000E600C7A006002","7A02611E38345800"
740 DATA "72023478011A4E92","55AE0058226E0058"
750 DATA "3384E80078037000","4E7570004E412E3C"
760 DATA "4449593149E8003C","2014670E2840BEAC"
770 DATA "000866F449EC000C","4E75588F70F94E75"
780 DATA "70F14E7561D4508C","601047EB0010BBCB"
790 DATA "62EE61C6588C6002","61C0347801124E92"
800 DATA "66E0554366DA3231","E8006BD448413231"
810 DATA "E8026BCC28814E75","202E009C67F82440"
820 DATA "327800E249EB0014","4E9166EA4881522C"
830 DATA "000E102C000E5300","672C080000006646"
840 DATA "342C000294416B10","322C0006B4416302"
850 DATA "3401394200026004","426C0002590066C8"
860 DATA "422C000E60C270F8","C001B03C008066F0"
870 DATA "7007C041103B0008","3940000C60AA0705"
880 DATA "030106040200302C","0000D0416B10322C"
890 DATA "0004B04163023001","39400000608A426C"
900 DATA "0000608400065345","523249520006FE94"
910 DATA "075054525F4F4646","FE22065054525F4F"
920 DATA "4E00FF280A505452","5F4C494D49545300"
930 DATA "FF28075054525F50","4F53FF1807505452"
940 DATA "5F4D4158FF000750","54525F494E430000"
950 DATA "0004FEA206505452","5F582500FE9C0650"
960 DATA "54525F592500FE58","07425554544F4E25"
970 DATA "FE7A0653594E4348","25000000","*",44348

```



# Neural Networking

**Dr Steven Wilcox's QL works on a new field of artificial intelligence.**

**T**his article aims to introduce a new type of computing that can only be simulated on a digital computer like the QL. This is neural computing, a relatively new area of research (in its current form) in the arena of artificial intelligence (AI). Artificial neural networks (ANN) resulted from an investigation into the usefulness of a set of simple processing units connected together in different ways. The result has been the this new form of computing, which has recently found its way out of the AI laboratory and into applications as diverse as stock market forecasting to speech recognition. Firstly, I would like to recap some of the short history of AI or, more correctly, machine learning.

Machine learning (ML) has played a central role in AI from the start of research into the field, although main thrust of AI research has deviated from this area with the arrival of such significant developments as problem solving, theorem proving, robotics and expert systems (ES). But recently ML has again started to play a significant and increasing part in the development of AI. For instance, early work on perceptrons (which are simple neuron-like processing units) suffered some setbacks because of an inability to solve non-linear functions, presented by Minsky and Papert [1]. However, there has been a resurgence in scientific and media interest due in part to the work of Hopfield [2], Ballard, Hinton and Sejnowski [3] who have played a large part in the development of ANNs with hidden layers (layers of neurons

that are not directly connected to the input or output of the network) capable of learning to recognise non-linear functions.

## How to learn?

The ability to learn, to adapt and to modify behaviour has been regarded by some as an exclusive property of organic beings, while some people argue that only human beings possess learning capabilities and intelligence. It is probable that the 'ability to learn' is a prerequisite for any form of intelligence and, in order to check for intelligence, a test has been proposed by Turing [4]. In this test a human being is asked to communicate, through a terminal, with two or more other operators one of which is the artificial system. If the human being can not tell whether the answers to his questions were from the system or the other human then the artificial system could be regarded as possessing true intelligence.

Early attempts at applying this test have proved how difficult it will be for a

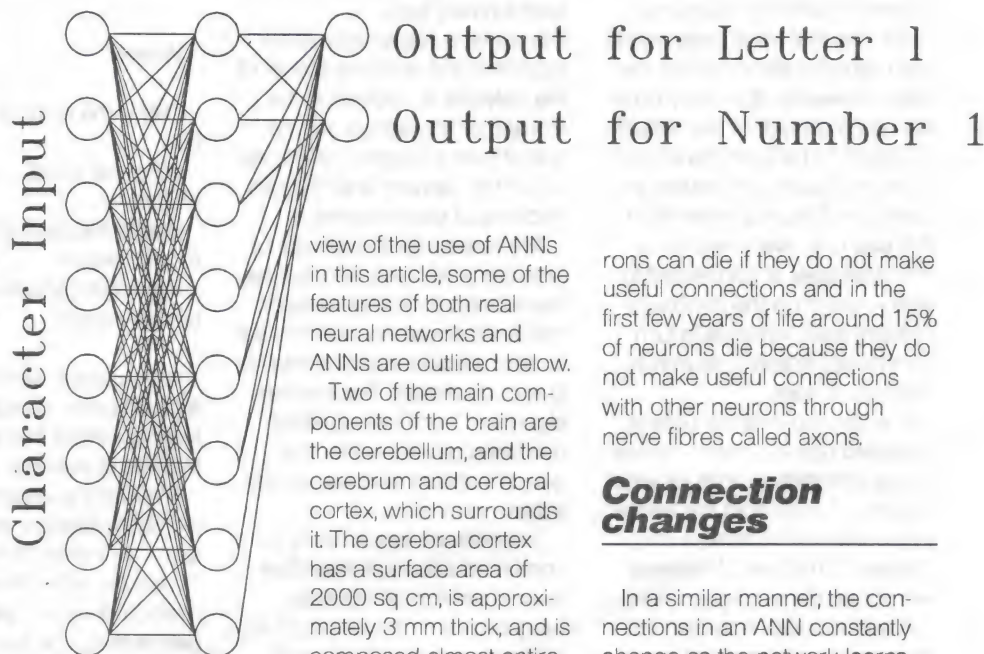
machine to achieve intelligence defined in this way. A computer program known as ELISA [5,6,7] achieved partial success until the person communicating with it worked out how it was composing its answers. In reality, the program does not 'understand' the words in the sentence but simply examined the sentence for key words to operate on, or used one of a set of standard replies.

## Neural networks

One approach to AI, as briefly mentioned earlier, is the artificial neural network (ANN). It is not clear whether the ANN approach is likely to achieve success as defined by Turing, but it has merits of its own where specific solutions are required to specific problems. In

about 100 billion neurons, with the central nervous system as a whole containing perhaps as many as 1000 billion. It is thought that the cerebrum and cerebral cortex perform most of the higher-order functions of our existence such as creativity and reasoning, with the cerebellum performing lower level functions such as the control of breathing.

A biological network is constantly changing as the organism learns new skills, and it is thought that this is partly achieved by changing the strength of the connections between neurons. As well as the strength of connections increasing and decreasing (the strength of the connection between two neurons is known as its "weight"), connections appear and disappear throughout the life of the organism. In fact, neu-



**Figure one: The neural network layout**

view of the use of ANNs in this article, some of the features of both real neural networks and ANNs are outlined below.

Two of the main components of the brain are the cerebellum, and the cerebrum and cerebral cortex, which surrounds it. The cerebral cortex has a surface area of 2000 sq cm, is approximately 3 mm thick, and is composed almost entirely of a layer of neurons six-deep. It contains

neurons can die if they do not make useful connections and in the first few years of life around 15% of neurons die because they do not make useful connections with other neurons through nerve fibres called axons.

## Connection changes

In a similar manner, the connections in an ANN constantly change as the network learns, and these too can appear and disappear as the strength of



connection between "neurons" increases or approaches zero.

The neuron makes connections with other neurons and communicates through the axon (the main communication fibre) and dendrite (the connection to the neuron). Communication is achieved by emitting electrical pulses (termed "firing"), around 2 milliseconds in duration and 100 millivolts in amplitude. When the sum of the input weights (the pulses from other neurons) the neuron receives exceeds a certain excitation level it will fire itself.

An artificial network doesn't emit pulses to communicate with other neurons. Instead, it emits a number between 0 and 1 that is a function of the inputs it receives from other neurons. If the sum of the inputs, when it has been weighted in a particular manner, exceeds a preset level then the neuron will emit the appropriate number. The precise procedure is described later in this article.

## ANNs

Artificial neural networks (ANNs) are a simplistic representation of this type of biological network. Usually they consist of a set of inputs with a series of hidden layers followed by a set of outputs. There are two broad classes of artificial neural network. The first class uses distributed representation, where the total knowledge (for every problem to be solved) of the network is distributed across the entire network. Figure one shows an example of such a network. In this example every neuron in the input layer is connected to every neuron in the middle (or hidden) layer, which is in turn connected to every neuron in the output layer.

The second method uses a localised representation, where single processing units, or small groups of units, hold the knowledge required to solve a single problem. This type of network will not be demonstrated here.

A network learns to discriminate between classes of patterns from an input domain in a holistic manner. The network is

presented with sets of features (usually one-dimensional arrays of data collected from the problem) that are representative of the instances of each class, correctly labelled, and the ANN is trained to recognise these. It does this by readjusting internal weights in a fixed network topology via a learning algorithm, so that the weights are optimised to the entire training set. These algorithms, in essence, calculate the amount of credit each active processing unit deserves by comparing the output of the system with the correct response.

Current ANNs do not accurately represent the biological equivalent, but rather exploit ways in which it is possible to connect a set of simple processing units. The application of the approach to real problems, such as word recognition, has shown that the ANN has the ability to recognise features more accurately from the input domain than has previously been possible using the Expert System-type approach [2].

## An ANN for the QL

The network that I'm going to present and use is of the simple feed-forward type, trained by a backpropagation algorithm that ensures that all of the network is involved in the storage of knowledge. Figure one shows a diagram of the layout of the network and the connections of each neuron.

The reason for this choice of network is relatively simple. The "feed-forward" type of network has been shown to perform well in other situations and is relatively well developed. The training algorithm, "back-propagation" has been developed for this type of network and has similarly been proven.

The network functions by the combined effects of individual neurons which compute a weighted sum of their inputs to give an excitation ( $\epsilon$ ). This is then used in a "sigmoid" function (the excitation function)

of the form:

$$f(\epsilon) = \frac{1}{1 + e^{-\epsilon}}$$

This type of function makes the neuron more "decisive" as the state of the neuron will switch from inactivity to activity very quickly around the central point of the function. For each neuron the training rule is to:

a) Present the network with the input. The neuron output is given by calculating:

$$o(i) = \sum w(j) \cdot i(j)$$

Where:

$o(i)$  is the output of the  $i$ th neuron

$w(j)$  is the weight of the  $j$ th input to the  $i$ th neuron

$i(j)$  is the input to the  $i$ th neuron on the  $j$ th input line

b) For each neuron compute the error, which is given by:

$$\Delta(i) = [o(i) - do(i)] \cdot f'(\epsilon)$$

Where:

$\Delta(i)$  is the error of the  $i$ th neuron

$o(i)$  is the actual output of the  $i$ th neuron

$do(i)$  is the desired output of the  $i$ th neuron

$f'(\epsilon)$  is the derivative of the excitation function

c) Adjust the weight  $w(i)$  by an amount given by  $k(\Delta(i) \cdot i(j))$ , where  $k$  is a constant that determines the rate of learning.

This training algorithm works for single layered networks but a bit more work is required for a multi-layer network as, this method does not allow the hidden layers to be trained. If these layers are ignored in the training algorithm then they will not assist in the decision making

process and are therefore redundant.

## Hidden layers

Hidden layers can be utilised by computing the error for each neuron in the output layer, ( $\Delta(i)$ ), and associating with each hidden unit in the layer below a proportion of the error of each output neuron to which it is connected. This proportion should obviously be the strength of the connection between the hidden neuron and the output neuron. This means that the hidden neuron's error will be a weighted sum of all the errors of all the output neurons to which it is connected, weighted again by the derivative of the excitation

function. This approach is then propagated back through the network, a layer at a time, until the input layer is reached. Hence the term "back-propagation".

If each input, as in Figure one, is used for character input and a number of different neurons are used to represent different recognised characters it is possible to train the network to recognise typed letters by the

state of its inputs (the letter) by repeatedly presenting random selections from the character set and telling the network which letter this is. If the training set is representative then the network can be presented with new data from the keyboard and asked to make a prediction as to which key the data comes from. The output of each neuron will then be representative of the probability of the letter being in that region. Obviously, the higher the probability the more certain of its choice of letter will the network be. If the network cannot decide on the letter then this will manifest itself as similar lower probabilities over several or all output neurons.

## SuperBasic ANN code

The ANN code and theory that I have presented is based on



the Hopfield (a researcher who has been very active in this area) network with a back-propagation training algorithm. The network is supervised during training, so is only of use where the supervisor knows what the output of the network should be for the training set of data. When the network has been trained, it then should be possible to present the network with new examples of the problem and with a little luck, and if the examples have been properly selected, the network should classify them correctly.

As an aside, a recent television programme contained a good example of a network that had not been trained on representative data. The example came from the US military where they had been training a network to recognise tanks in various types of undergrowth. They had tanks in open fields, partly hidden by bushes, groups of tanks, etc. The training went well, but when they came to test the network on examples that had not been presented previously, the network performed very poorly. After a small investigation it was found that the training data consisted of slides of tanks on sunny days and slides of no tanks on cloudy days - the network had learnt to recognise the difference between a nice and a grotty day, a possibility that had none of the intelligent life forms present had taken into consideration! This emphasises the need to make the training data representative of what you want the network to learn.

In order to train the network, large numbers of presentations of each input have to be made. In this example around five hundred is the minimum that can be used. All this takes some time, and if you have a compiler then everything will be speeded up enormously. The code has been compiled with QLiberator 3.35, so should compile with little problem with other compilers, although I have not been able to test this. It should also be possible to run the network on a basic 128K QL as there are no large arrays and the code is not very big.

## Setting weights

The code consists of five procedures, 'run\_network', 'get\_input', 'setup\_network', 'show\_results' and 'train\_network' with 'run\_network' and 'train\_network' being the main routines. Procedure 'setup\_network' simply sets the weights of every connection to a random value between -0.5 and 0.5 so that the weights are set to values that are not the same every time the network runs. Procedure 'get\_input' randomly selects either the letter 'I' or the number '1' from the character set, then determines the equivalent binary value (input vector) and the correct output for each output neuron. (The training flags in the array target() is changed from 1 or 0 to 0.9 and 0.1 as it is very difficult for the weighting function used to approach 1 and 0.)

Procedure 'run\_network' takes the input vector and with the current weights calculates the output of the network, which is then used in the procedure 'train\_network' to propagate the error derivative backwards through the network from output to input whilst adjusting the weights to minimise the error. Finally, procedure 'show\_results' crudely displays the results either while training is taking place or when the network is being tested after training is completed.

When you first train the network it can appear to be a lost cause, as the error takes quite a time to reduce, at the start it may even increase for awhile. However, if you persevere and come back in the morning, the training will be completed and you will be able to test it with new input from the keyboard. This is where lines 1220 to 1600 come in as they take the code for the letter 'I' or the number '1', or a random pattern if you press any other key, and present it to the network. Strictly speaking this is not a proper test of the network as the test data and the training data are identical and, it would be very disappointing if the network failed. This is however just an

```

1010 CLS
1020 output_neurons=2
1030 input_neurons=8
1040 layers=2
1050 iterations=500
1060 DIM neuron_output(input_neurons,layers)
1070 DIM neuron_weight(input_neurons,
input_neurons+1,layers)
1080 DIM delta(input_neurons,layers)
1090 DIM inputs(input_neurons),
network_output(input_neurons),target(input_neurons)
1100 setup_network_weights layers,input_neurons
1110 count=0
1120 average_error=0
1130 REPEAT train_network_loop
1140   get_input input_neurons,output_neurons
1150   run_network output_neurons,input_neurons,
layers
1160   train_network output_neurons,input_neurons,
layers,.7,neuron_error
1170   count=count+1
1180   show_results output_neurons,input_neurons,
neuron_error,count,average_error,1
1190   IF count>=iterations THEN EXIT train_net
work_loop
1200 END REPEAT train_network_loop
1210 :
1220 REPEAT get_new_data
1230   CLS
1240   PRINT "Press I,1 or any other key"
1245   CLS
1250   PRINT
1260   key$=INKEY$(#0,-1)
1270   key=CODE(key$)
1280   SELECT ON key
1290     =49: RESTORE 1590
1300     FOR i=1 TO input_neurons
1310       READ inputs(i)
1320     END FOR i
1330     run_network output_neurons,input_neu
rons,layers
1340     show_results output_neurons,input_ne
urons,neuron_error,count,average_error,0
1350     =108: RESTORE 1600
1360     FOR i=1 TO input_neurons
1370       READ inputs(i)
1380     END FOR i
1390     run_network output_neurons,input_neu
rons,layers
1400     show_results output_neurons,input_neu
rons,neuron_error,count,average_error,0
1410   =REMAINDER :
1420     a=RND:b=RND:c=RND:d=RND:e=RND:f=RND:
g=RND:h=RND
1430     IF a>.5 THEN inputs(1)=1: ELSE
inputs(1)=0: END IF
1440     IF b>.5 THEN inputs(2)=1: ELSE
inputs(2)=0: END IF
1450     IF c>.5 THEN inputs(3)=1: ELSE
inputs(3)=0: END IF
1460     IF d>.5 THEN inputs(4)=1: ELSE
inputs(4)=0: END IF
1470     IF e>.5 THEN inputs(5)=1: ELSE
inputs(5)=0: END IF
1480     IF f>.5 THEN inputs(6)=1: ELSE
inputs(6)=0: END IF
1490     IF g>.5 THEN inputs(7)=1: ELSE
inputs(7)=0: END IF
1500     IF h>.5 THEN inputs(8)=1: ELSE
inputs(8)=0: END IF
1510     run_network output_neurons,input_neu
rons,layers
1520     show_results output_neurons,input_neu
rons,

```



```

neuron_error,count,average_error,0
1530 END SElect
1540 PRINT
1550 PRINT "Press any key to continue"
1560 a$=INKEY$(#0,-1)
1570 CLS
1580 END REPEAT get_new_data
1590 DATA 0,0,1,1,0,0,0,1
1600 DATA 0,1,1,0,1,1,0,0
1610 :
1620 DEFine PROCedure run_network(output_neurons,
input_neurons,layers)
1630 FOR i=1 TO input_neurons
1640 e=neuron_weight(i,input_neurons+1,1)
1650 FOR j=1 TO input_neurons
1660 e=e+neuron_weight(i,j,1)*inputs(j)
1670 NEXT j
1680 neuron_output(i,1)=1/(1+EXP(-e))
1690 NEXT i
1700 :
1710 IF layers>1 THEN
1720 FOR k=2 TO layers
1730 FOR i=1 TO input_neurons
1740 e=neuron_weight(i,input_neurons+1,k)
1750 FOR j=1 TO input_neurons
1760 e=e+neuron_weight(i,j,k)*neuron_out
put(j,k-1)
1770 NEXT j
1780 neuron_output(i,k)=1/(1+EXP(-e))
1790 NEXT i
1800 NEXT k
1810 END IF
1820 :
1830 FOR i=1 TO output_neurons
1840 netork_output(i)=neuron_output(i,layers)
1850 NEXT i
1860 END DEFine run_network
1870 :
1880 :
1890 DEFine PROCedure get_input(input_neurons,
output_neurons)
1900 a=RND
1910 IF a>=.5 THEN
1920 RESTORE 2140
1930 FOR i=1 TO input_neurons
1940 READ inputs(i)
1950 END FOR i
1960 target(1)=1
1970 target(2)=0
1980 END IF
1990 IF a<.5 THEN
2000 RESTORE 2150
2010 FOR i=1 TO input_neurons
2020 READ inputs(i)
2030 END FOR i
2040 target(1)=0
2050 target(2)=1
2060 END IF
2070 FOR i=1 TO output_neurons
2080 IF target(i)=1 THEN
2090 target(i)=.9
2100 ELSE
2110 target(i)=.1
2120 END IF
2130 END FOR i
2140 DATA 0,1,1,0,1,1,0,0 : REMark 1
2150 DATA 0,0,1,1,0,0,0,1 : REMark 1
2160 END DEFine get_input
2170 :
2180 :
2190 DEFine PROCedure setup_network_weights(
input_neurons)
2200 FOR k=1 TO layers
2210 FOR i=1 TO input_neurons

```

example and, for more general input the procedure 'get\_input' could be altered to take training data from a file that contained scanned images for example.

## Conclusions

After, all this you might be thinking 'so what', there are excellent routines for getting keyboard input and turning it

into a character on the screen, but this misses the point. This code has learnt to recognise the two characters by being presented with examples, and so in principle it should be possible to extend the network so that it recognises more characters, and even characters from other sources, such as a scanner, or a paint package. Perhaps you could try scanned images of the

```

2220 FOR j=1 TO input_neurons+1
2230 neuron_weight(i,j,k)=(RND-.5)
2240 NEXT j
2250 NEXT i
2260 NEXT k
2270 END DEFine setup_network_weights
2280 :
2290 :
2300 DEFine PROCedure show_results(output_neurons,
input_neurons,neuron_error,count,average_error,
print_target)
2310 AT 0,0
2320 PRINT "A Simple Neural Network"
2330 FOR i=1 TO input_neurons
2340 PRINT "Input ";i;inputs(i)
2350 END FOR i
2360 PRINT
2370 IF print_target=1 THEN
2380 FOR i=1 TO output_neurons
2390 PRINT "Target ";i;" ";target(i);"Result
";i;" ";netork_output(i)
2400 END FOR i
2410 PRINT
2420 END IF
2430 IF print_target=0 THEN
2440 FOR i=1 TO output_neurons
2450 PRINT "Result ";i;" ";netork_output(i)
2460 END FOR i
2470 PRINT
2480 END IF
2490 PRINT "Number of Itererations ";count
2500 PRINT "Error this time ";neuron_error
2510 average_error=.2*neuron_error+.8*
average_error
2520 PRINT "Smoothed error ";average_error;
" "
2530 END DEFine show_results
2540 :
2550 :
2560 DEFine PROCedure train_network(output_neu
rons,
input_neurons,layers,eta,neuron_error)
2570 neuron_error=0
2580 FOR i=1 TO input_neurons
2590 IF i<=output_neurons THEN
2600 delta(i,layers)=(target(i)-netork_output
(i))*netork_output(i)*(1-netork_output(i))
2610 neuron_error=neuron_error+(target(i)-
netork_output(i))*(target(i)-netork_output(i))
2620 ELSE
2630 delta(i,layers)=0
2640 END IF
2650 NEXT i
2660 FOR k=layers-1 TO 1 STEP -1
2670 FOR i=1 TO input_neurons
2680 deltasum=0
2690 FOR j=1 TO input_neurons
2700 deltasum=deltasum+delta(j,k+1)*
neuron_weight(j,i,k+1)

```



letter 'l' and number '1' from source code printouts, as these are often difficult for humans to distinguish.

Whatever you do, please experiment with the network and feel free to modify and use it wherever you can. The area of AI, and in particular that of neural networks, is really only just developing, so this is an area where individuals can make progress in the field. There are even some text books appearing, which any decent library will be able to order (see Looking for a Book?, Hardy Hints, QL World March 1993) and so it is relatively easy to 'bootstrap' yourself to the current state of understanding, and then you may be able to push forward the frontier of human knowledge.

## References

**M.Minsky and S.Papert**,  
Perceptrons: an Introduction to Computational Geometry MIT Press, Massachusetts, 1969.

**D.W.Tank and J.J.Hopfield**,  
Neural Computation by Concentrating Information in Time Proceedings of the National Academy of Science, 84, April, 1987, pages 1896-1900.

**D.Ballard and G.E.Hinton and T.J.Sejnowski**, Parallel Visual Computation Nature, 306, 21, page 26, 1983.

**A.M.Turing**, Computing Machinery and Intelligence Mind, 59, pages 433 460, 1950.

**J.Weizenbaum**, ELIZA - A Computer Program for the Study of Natural Language Communication between Man and Machine Communications of the Association for Computing Machinery, 9, pages 36 - 45, 1966.

**J.Weizenbaum**, Automating Psychotherapy, ACM Forum, 17, page 543, Reprinted with Replies, CACM, Volume 26, page 28, 1983.

**J.Weizenbaum**, Computer Power and Human Reason: From Judgment to Calculation, W.H.Freeman, San Francisco, 1976.

```

2710 NEXT j
2720 delta(i,k)=neuron_output(i,k)*(1-
neuron_output(i,k))*deltasum
2730 NEXT i
2740 NEXT k
2750 FOR i=1 TO input_neurons
2760 FOR j=1 TO input_neurons
2770 neuron_weight(i,j,1)=neuron_weight(i,j,1)+
eta*delta(i,1)*inputs(j)
2780 NEXT j
2790 neuron_weight(i,input_neurons+1,1)=
neuron_weight(i,input_neurons+1,1)+eta*delta(i,1)
2800 NEXT i
2810 :
2820 :
2830 FOR k=2 TO layers
2840 FOR i=1 TO input_neurons
2850 FOR j=1 TO input_neurons
2860 neuron_weight(i,j,k)=neuron_weight(i,j,k)+
eta*delta(i,k)*neuron_output(j,k-1)
2870 NEXT j
2880 neuron_weight(i,input_neurons+1,k)=neu
ron_weight(i,input_neurons+1,k)+eta*delta(i,k)
2890 NEXT i
2900 NEXT k
2910 END DEFine train_network

```

*Now available on a QL near you...*

# GT-Prolog/QL


**Edinburgh Syntax  
Incremental Compiler  
Automatic Garbage Collectors  
Tail-Recursion Optimisation  
First Argument Indexing/Hashing  
32bit integers / 48bit reals  
255 byte atoms / 32kb strings**

**Interactive Workbench (needs 512k)**  
\* Windows, Menus, Dialogues  
\* Editor  
\* Debugger  
**User Guide, Reference Manual  
Full QDOS compatibility  
Configurable Memory up to 16Mb**

**Turbo Performance - Naive Reverse exceeds 5k LIPS (GC)**

**Price including postage/packing £89.95**

For more information contact us at  
**Rosebank, Stream Road,  
Upton, Oxon, OX11 9JG  
Tel/Fax: 0235-851818**

 **Grange  
Technology  
Limited**



# ProPub Toolbox

**Bryan Davies  
sizes up Digital  
Precision's  
ProPub assis-  
tants.**

## INFORMATION

Program: Professional  
Publisher Toolbox I & II

Price: £29.95 each,  
£49.95

for both

Supplier: Digital Precision  
Ltd., 222 The Avenue,  
Chingford, London  
E4 9SE.

Tel. 081 527 5493.

## ProPub

Toolbox I was reviewed in early 1991, and is essentially the same now as then (including the price). It is worth looking at it again because the interest in DTP is, if anything, higher now than it was, and because there is a Toolbox II to go with it.

Reviewers have an unending stream of new programs (and new versions of existing programs)

placed before them. It's easy to forget that many users stick with the same program - maybe even the same version - for years. These users welcome add-on programs, which extend the range their existing programs without creating headaches. Learning new main programs usually means days or months of study, whereas add-on items can quickly be mastered.

Both Toolboxes come on floppy disk only, and require a minimum 256 KB memory expansion. As ProPub itself requires a minimum 512 KB extra, the buyer of a Toolbox is almost certain to have a suitable system.

## Toolbox I

There are two programs, a set of 44 founts, a collection of 13 clip art images, and a few odd demonstration graphics files on the disk. There is also a boot file and a SuperBasic extensions file. These two files will normally be incorporated into the user's ProPub boot file, as the boot is a one-liner which loads the extensions, and these are included in the Toolkit extensions file with ProPub (and in the Xtras file with The Editor). There is an UPDATES.DOC file, which is short and points out that some items noted in the instructions are no longer on the Toolbox I disk, but incorporated into Toolbox II; in their place, there are extra clip art files.

The founts all have the suffix .HDF, indicating that they are the better, high-definition type. The clip art files number thirteen and have the suffix .CUT, which will probably be familiar to regular DTP users. The other graphics files are an assortment of Digital Precisions' own files, and have been seen elsewhere (such as with ProPub itself).

The clip art files are not

sensational, but could be useful if you require illustrations of that type; various animal and country images, for instance. The founts are the better part of the disk, covering quite a wide range of styles.

The programs are utilities - Grab\_Sector\_ClipArt and PrepText. Sector Software had a collection of clip art files, and Grab is designed to capture these images and put them into a form suitable for loading into ProPub: standard 32 KB screen dump format. PrepText is aimed at text created in a word-processing program, the obvious one here being Perfection. It puts text into a more suitable shape for loading into ProPub; untreated text can show oddities after import. The text should be saved in Ascii form - that is, without the formatting information used by the word-processing program. PrepText keeps basically the same format but makes changes to maintain the correct line format.

## Toolbox II

The instructions for Toolbox I were stated to be the "shortest manual" DP has issued - one page. Toolbox II has two pages so, by DP standards, it too is short. This second disk contains 38 more .HDF founts, 6 .QLS founts (which are not of quite the same high quality), and 4 .FND founts. There are three utility programs - PPCompressor, PPPSaver and QWConverter.

PPCompressor is a space-saver for ProPub page files. The stated size range for standard page files is 28,800 to 320,000 bytes, and it is easy to fill a few disks. Apart from compressing the files, this program can put them onto another device, thereby performing a back-up function. As ProPub cannot load compressed files,

PPCompressor has an expand function, too.

A sample compressed file is supplied, and the user can expand it with PPCompressor and check it with PPPSaver. This program enables part of a ProPub page to be saved, for use in graphics programs or with SuperBasic.

The third program gets its name from Qwriter II. Founts from that program can be converted to ProPub .HDF form by QWConverter; they can also be enlarged with another DP program, Font Enlarger. The fount files must have the suffix .FND, but Qwriter II itself has a converter, to change .FNT files to .FND.

All three programs can be either EXEC-ed or loaded as SuperBasic extensions. In the second case, they are activated simply by typing (for example) PPCOMP ENTER. That is, the file PPCOMPRESSOR can be handled in these ways:

```
EXEC.W "flp1_PPCOMPRESSOR"
```

or

```
base=RESPR(50*512):LB  
YTES "flp1_PPCOMPRESSOR",base:CALL base
```

Minerva users can run these programs on the second screen.

## In Use

One of the standard sample files with ProPub is called DEMO\_PAGE, and its normal size is 96018. After treatment with PPCompressor, the size dropped to 50582 bytes, roughly 53% of the initial size. The compression operation took one second! Expanding the compressed file brought it back to its full size, in slightly longer time. The program misleadingly stated that the operation had been a fur-



ther "compress"; in fact, trying a further compression operation produced no change in file size, so the initial pass looks to be fully effective.

The expanded file was looked at with PPPSaver. It came up on the screen quickly and correctly, but without the menu that the instructions state should be displayed. That created no problem, as you can follow the written instructions for positioning the page, then press Space to indicate that you have selected the area you want. The program creates 32 KB files, and that clearly means you can select only a full screen for saving. The save process took a second or two. Escaping from the program was not a clean process, as the image of the expanded page stayed on the screen.

Both the compressed and expanded page, and

the saved part of that page, loaded into ProPub properly. DTP is space-hungry. If you can save space of this order, it will save you money in disk costs; files you value should have at least one back-up copy, so the saving is two-fold.

## Conclusion

With programs such as these there is no question of comparing them with competitive products and deciding which is the best to buy. For the most part, there is no third-party competition in the QL world. If you are a serious ProPub user, there will be something of interest here, and it is simply a question of whether or not your budget runs to the purchase price at the time. We are into the Christmas season now, and you might use a few extra founts to decorate the tree...

Telephone: 0753-888866  
Fax: 0753-887149

(EEC)

W.N. Richardson & Co.

18-21, Milbourne House,  
Chalfont St. Giles, Bucks HP8 4UE

## SINCLAIR QL PRICE LIST MARCH '93

All Prices Include 17.5% VAT

### QL Computers NEW MONITOR £50 CHEAPER

JS COMPLETE QL Computer, PSU, T.V. Lead, PSION V2.35 Software (Word Processor, Spreadsheet, Database & Business Graphics), Handbook for QL Programs, & Superbasic, JS £120  
Free one year membership to QUANTA, the Independent QL User Group with over 400 free library programs, Newsletters & HELP. 6 MONTHS WARRANTY. WITH 1M ROM £100.00

PART EXCHANGE £30 OFF ABOVE. Send in old QL (Unit only, any condition) - JS ROM £90 JM £70.00

BACKUP DISK QL & PSU only. JS £80 JME65 (Part exchange allowance £15 if required)

### Accessories

\* NOTE: EXTERNAL (SER2) 3 BUTTON MOUSE AND SOFTWARE WITH EXTRA FUNCTIONS. NOW HERMES COMPATIBLE \*

PC KEYBOARD INTERFACE, INTERNAL FITTING, FOR FITTING 102 KEY KEYBOARD £75.00  
PC KEYBOARD, UK version, 102 key (AT) £30.00  
PC KEYBOARD INTERFACE AND KEYBOARD (INCLUDES FREE JOYSTICK OR PSU) £95.00  
CABLE AND LEAD FOR EXTERNAL FITTING of Keyboard Interface £14.00  
JOYSTICK with QL lead (no interface required) £10.00  
\* QL MOUSE 3 Button, Software controlled, externally mounted, simply fits in SER2 £45 CORDLESS MOUSE £55.00

### Disk Drives

THE UNIVERSAL DRIVES ARE ALSO SUITABLE FOR ACORN BBC, ATARI ST, AMIGA, SPECTRUMS, IBM, AMSTRAD AND OTHER PC COMPATIBLES

Universal 3.5" 1Mb disk drive, cased with PSU and free QL lead £70.00 2 Units for dual drive £120.00  
Universal 3.5" 2Mb disk drive, as above but 2Mb capacity £90.00 2 Units for dual drive £170.00  
3.5" 1Mb Uncased disk drive £25.00  
3.5" 2Mb Uncased disk drive £49.00  
Lead for uncased disk drives, or other universal micros £10.00 POWER SUPPLY FOR UNCASED DRIVES £6.00  
METAL CASES FOR UNCASED DRIVES £6.00

### Parallel Printers

### NEW RANGE

SAMSUNG SP093 80 Col, 300 cps, 50 NLQ, 3K Print Buffer, Centronics, Tractor Feed/Sheet Feed, Paper Park £149.00  
OLIVETTI COLOUR PRINTER, 24 pin, 200 cps, 50 NLQ, 40K Buffer, Epson LQ2550/IBM Compatible £259.00  
CENTRONICS PARALLEL Printer Interface for above printers £7.00  
QL LEAD FOR SERIAL PRINTERS £10 CANNON BJ10 EX £275

### Monitors

PHILIPS 14" HIGH RES COLOUR EGA 31 DOT PITCH, FULL 85 COL WITH AMBER OR GREEN TEXT FEATURE, IDEAL FOR WORD PROCESSING, RECONDITIONED, 90 DAY WARRANTY. £149

OK FOR PC AND QL SELF SENSING AND MANY EXTL CONTROLS. £39

### Microdrive Cartridges and Spares

### "MONO OPTION" WITH 9" GREEN SCREEN

4 New Cartridges in a wallet £10.00 8 prog carts for reformatting in 2 wallets £15.00  
Plastic Storage filing box including 20 new cartridges £45.00  
QL Psion Software V2.35 Includes Quill, Abacus, Archive, and Easel IN WALLET £18.00  
QL Psion Software Separate programs £10.00  
QL power Supply Unit £10.00 Membrane (and instructions) £9.00  
TV or Network leads £3.00 QL Top & Bottom Case £5.00  
IC's ZX £301 £6.00 ZX £302 £3.00 8049 (IPC) £3.00 MC 1377 £3.00  
MC 68008 £12.00  
QL SERVICE MANUALS & CIRCUIT £25.00 S.A.E. FOR FURTHER DETAILS

Payment terms:  
CWO, Access, VISA et cetera  
Cheques - allow 10 days

Delivery:  
Carriage - £9  
Postage - £5

Product subject to availability, E&OE.

TEL: 0753 888866 FAX: 0753 887149

# ARCHIVE YOUR QL COLLECTION

Now you can keep your Sinclair QL World magazines safe and clean. No more dog-eared covers or missing copies... You can protect your magazines in this high quality, specially-created binder. This Sinclair QL World binder will comfortably hold a complete year's issues of your favourite Sinclair magazine. It is a high

quality product, British-made and comes with full binding instructions. It is manufactured in a rich, deep blue with genuine gold blocked lettering. Enhance your Sinclair QL World Magazine collection now for only £6.95 (inc. P&P!) Send for one today! The QL World binders also make an ideal gift for other Sinclair users too!

TO: QL WORLD, THE BLUE BARN, WOOTTON, WOODSTOCK, OXON. OX7 1HA

Please send me ☐ QL World binders - I enclose £6.95 for each binder including VAT: postage & packing.

Readers outside the UK and Eire please add £1.95 for surface overseas mail.

Please make cheques payable to ARCWID LTD.

ACCESS ☐ VISA ☐ account:

Expiry date:

Name

Address

Postcode

Tel No.





# SUBSCRIBE TO SINCLAIR QL WORLD AT LAST YEARS PRICES AND SAVE £££££££'s

£££££

**UK**  
**£23.40**  
**(12 MONTHS)**

**EUROPE**  
**£32.90**  
**(12 MONTHS)**

**REST OF  
WORLD**  
**£40.90**  
**(12 MONTHS)**

PLEASE SEND YOUR  
SUBSCRIPTION FORM TO  
SINCLAIR QL WORLD,  
THE BLUE BARN, TEW  
LANE, WOOTTON,  
WOODSTOCK, OXON.  
OX71HA.  
TEL:0993 811181  
FAX:0993 811481

I WISH TO SUBSCRIBE TO  
SINCLAIR QL WORLD  
WE HAVE HAD TO RAISE THE  
COVER PRICE OF QL WORLD  
BUT WILL BE HOLDING LAST  
YEARS SUBSCRIPTION PRICE  
FOR THE IMMEDIATE FUTURE  
ONLY.

**Existing subscribers will not be affected.**

.....  
PLEASE DEBIT MY CREDIT CARD ACCESS/VISA

CARD NUMBER.....

EXPIRY DATE.....

PLEASE SEND ME A 12 MONTH SUBSCRIPTION TO SINCLAIR  
QL WORLD.

NAME.....

ADDRESS.....  
.....

TELEPHONE NUMBER.....

SIGNED.....

DATE.....

STARTING WITH ISSUE.....



**SINCLAIR QL** + twin floppy drives, 256k memory, separate keyboard, many games £125 0491 681821

**QL SYSTEM** 2XQL's, Colour Monitor, 2x31/2" Disc Drives, Trumpcard, Serial 8056 Printer, Silver-reed plotter/printer. Psion l1cm organiser, comms-link, Datapaks, Letter Organiser. QL s/w. taskmaster, Spellbound, Lightning, Techniq, Finance etc. All Leads & Manuals. Buyer to collect £375 Tel: 051 726 9801 (eve).

**QL unexpanded AH** version PSU Psion software manual etc. Any offers Miracle Systems QL modem £20. Phone: John 0689 855444

**JS QL, Metal** case, PC keyboard and I/face, twin 3-1/2" d/drives and OPD m/drives, some software, new colour monitor £250 ono. Also Tandata modem 3, CQVI Digitiser and more. Tel: (Derby) 0332 677746

**QL unexpanded JM version**, shinwa CPA 80Q printer, joystick, Software including Psion suite, Eye-Q and Games, 40 MDV's, Manuals etc, £150 Tel:0256 764006 evenings only

**GIVEAWAY!** Parting with old friends. For only £75 plus carriage you get 2QL's (Ver JM), Dual disk drive, Trump Card, Interface, PC Conqueror, Accelerator, Doctor, Othello, and much more! Sheffield 0742 680019

**WITH REGRET**, Offers for complete QL System. Two Computers 14" colour monitor, double disk drive, ice, original software plus more. Books, Manuals. Psion organiser and printer. Phone for details 0257 265700

# MICRO ADS

## Professional poolster REVOLUTIONARY NEW SYSTEM

### COMBINES

Sound logic with A.I. to win £ hundreds £  
Analysis of Bookies Factors for matches  
Common sense improves chances of Mega-Win  
UNIQUE ENTRY SYSTEM

Takes only a few minutes each week  
Over 300 teams in data-bank.

Minor league teams automatically added.  
Prior to PC release @ £99.00

To QL users offered at SPECIAL PRICE £49

### THIS MONTH

**MONEY BACK GUARANTEE** - Make a profit in your first full season or your money back.

256K ROM minimum. Discs Only. State 3.5 or 5.25

**CHEQUE OR P.O. TO S.CANTON, OCEAN COMPUTER SERVICES, PO**

**BOX 68, NORTH DO. Manchester M8 6NJ**

**Tel. 061 740 9002**

Please send coupon to: Advertising Department Sinclair/QL World, The Blue Barn, Wootton, Woodstock, Oxon. OX7 1HA. Please place this Classified Advertisement in the next issue of Sinclair/QL world.

NAME .....

ADDRESS .....

DAYTIME TELEPHONE NO .....

I enclose a cheque/P.O. to the value of £17.65 (inc. VAT) for a lineage advertisement of no more than 30 words. Please make cheques payable to: Arcwind Ltd.


# CLASSIFIED



# DJC

# Dilwyn Jones Computing

41 BRO EMRYS, TAL-Y-BONT, BANGOR,  
GWYNEDD, LL57 3YT, GREAT BRITAIN  
FAX/TEL: (0248) 354023

## DATABASE SOFTWARE

COCKTAILS WAITER	£10.00
DATA DESIGN 3	£60.00
DATA DESIGN API	£20.00
DBEASY	£15.00
DBPROGS	£15.00
FLASHBACK	£25.00
FLASHBACK SE	£40.00
SUPER DISK INDEXER	£12.00

## DTP & CLIPART

PAGE DESIGNER 3	£40.00
Available at last!	
UPGRADE FROM PD2	£25.00
SCANNED CLIPART 1 (U)	£10.00
SCANNED CLIPART 2 (U)	£10.00
THE CLIPART (U)	£12.00

## FILE HANDLING

FILEMASTER	£12.00
FILES 2 (U)	£12.00
4MATTER & LOCKSMITHE	£23.50
LOCKSMITHE (U)	£14.95
MDV TOOLCHEST	£14.95
THE GOPHER (U)	£12.00
WINBACK 2	£25.00

## FILE TRANSFER

CONVERT-PCX	£10.00
DISCOVER	£20.00
MULTI-DISCOVER	£30.00
OPD INTERCHANGE	£15.00
QL-PC FILESERVER	£24.50
STOQL	£24.50
TEXTIDY	£15.00

## GAMES & LEISURE

FIVE GAME PACK (U)	£12.50
FLEET TACTICAL COMMAND (U)	£39.95
FTC DATA PRINTER	£9.95
FLIGHTDECK (U)	£15.00
FUGITIVE	£9.95
GREYWOLF	£12.50
OPEN GOLF	£12.50
QUESTION MASTER (U)	£10.00
QUIZ MASTER 2 (U)	£12.50
RETURN TO EDEN	£17.50
SOLITUDE (U)	£15.00
SQUIDGY ROUND THE WORLD (U)	£12.50

## LABELLING

ADDRESS BOOK & LABEL PRINTER	£15.00
SUPER DISK LABELLER	£10.00

## GENEALOGY

GENEALOGIST 3	£60.00
new pointer driven version	
Upgrade 2nd Edition	£33.00
upgrade other versions, ask	
GENEALOGIST 2ND EDITION	£30.00
BUDGET 128K GENEALOGIST (U)	£12.00

## GRAPHICS

IMAGE-D (U)	£10.00
IMAGE PROCESSOR	£15.00
LINE DESIGN	£100.00
PAINTER	£25.00
PICTUREMASTER	£15.00
PICTUREMASTER PLUS	£20.00
QRCTAL	£20.00
QUICK MANDELROT 3 (U)	£15.00
SCREEN COMPRESSION	£10.00
SCREEN SNATCHER (U)	£10.00
SIMPLE VIDEO TITLES	£5.00
TRANS24 (U)	£10.00

## HARDWARE

MICRO PROCESS CONTROLLER	
new redesigned version	£64.50
MPC SOFTWARE	£9.95
NETWORK PROVER	£4.00
QPOWER REGULATOR	£24.95
(unsuitable Gold Card)	
SERMOUSE	£40.00
VOICE ANALYSER-New!	
Ask for details and price	
Add £2.50 postage for MPC,	
SERMOUSE AND VOICE ANALYSER	

## MAGAZINES

QREVIEW	£2.00
QL TECHNICAL REVIEW	£2.00
issues 1-8 available	
QL ADVENTURER'S FORUM	£1.75
Issues 1-9 available	
QL LEISURE REVIEW	£2.00
issues 1-2 available	
(UK prices only shown)	

## PROGRAMMING

BASIC REPORTER (U)	£10.00
DISA	£29.00
DJTOOLKIT (U)	£10.00
EASYPTR 3 PART 1	£40.50
EASYPTR 3 PART 2	£20.00
EASYPTR 3 PART 3	£20.00
MEGATOOLKIT DISK (U)	£25.00
MEGATOOLKIT EPROM (U)	£40.00
QLIBERATOR 3 36	£50.00
BUDGET QLIBERATOR (U)	£25.00
QLOAD AND QREF (U)	£15.00
S EDIT EDITOR	£20.00

## SCREEN DUMPS

SIDEWINDER PLUS	£24.95
-----------------	--------

## SPREADSHEET PRINTING

SIDEWRITER (U)	£15.00
3D TERRAIN	£12.50

## SUNDRIES

3 5" DSDD DISKS each	£0.40
3 5" DSHD DISKS each	£0.70
3 5" DISK LABELS (roll 100)	£2.00
3 5" LABELS (printer roll)	£2.50
ADDRESS LABELS (roll 100)	£2.00
MICRODRIVE CARTRIDGES	£2.50
MDV LABELS (roll 100)	£2.00
MOUSE MAT	£2.50
3 5" DISK DIVIDERS (20)	£3.00
Add £0.50 postage for labels or mouse	
mat if only ordering those, or £2.50 postage	
for disks or disk box divider sets	

## TEXT

BANTER BANNERS	£25.00
BIBLE TEXT	£20.00
QTP2	£29.95
QUICK POSTERS (U)	£10.00
ROB ROY PACK (U)	£10.00
SPELLBOUND	£30.00
SPELLBOUND S E	£50.00
TEXTIDY	£15.00
TEXT 'N' GRAPHIX	£20.00

## OTHER SOFTWARE

HOME BUDGET (U)	£20.00
PRINTERMASTER (U)	£20.00
QPAC1	£19.95
QPAC2	£39.95
THE Pointer Environment package!	
QTOP	£29.95
SCREEN DAZZLER	£15.00
SCREEN ECONOMISER (U)	£10.00
SLOWGOLD (U)	£5.00
SPEEDSCREEN (U)	£15.00
SPEEDSCREEN EPROM (U)	£30.00
TASKMASTER	£25.00
VISION MIXER 1	£10.00
VISION MIXER PLUS	£22.50

CALL OR WRITE FOR A FREE COPY OF  
OUR QL SOFTWARE CATALOGUE  
DETAILING AROUND 100 QL PRODUCTS

PLEASE NOTE: (U) ABOVE MEANS THAT  
THE SOFTWARE IS SUITABLE FOR USE  
ON A 128K QL



TERMS: POSTAGE-software sent post free to UK, overseas add £1.00 per program (maximum £3.00) Floppy disks, SERmouse, etc add £2.50 postage (see above) PAYMENT - IN UK currency (pounds sterling) only, please Valid methods of payment are cheque drawn on UK branch of bank or building society, Eurocheque, Postal Order, cash (send by registered post) or by credit card - Visa, Access, Mastercard and Eurocard accepted Please make cheques etc payable to DILWYN JONES COMPUTING Minimum order value £5.00 (due to bank charges) Goods remain property of Dilwyn Jones Computing until paid for in full Orders normally sent out within a few days of receipt, we will attempt to advise if any delay anticipated due to stock problems Orders can be accepted by telephone if paid for with credit card Orders paid with credit card can only be sent to cardholder's address under card company rules FAX - We now have a Fax machine on our usual number (0248) 354023

Master Card

Euro Card